

参赛密码 \_\_\_\_\_  
(由组委会填写)

**第十二届“中关村青联杯”全国研究生  
数学建模竞赛**

学 校 海军工程大学

---

参赛队号 90038010

---

1.冯 杭

---

队员姓名 2.杨忠超

---

3.刘 璐

---

参赛密码 \_\_\_\_\_  
(由组委会填写)



## 第十二届“中关村青联杯”全国研究生 数学建模竞赛

题 目      面向节能的单/多列车优化决策问题

摘                  要：

近年来针对减少列车牵引能耗的列车运行优化控制问题成为轨道交通领域的重要研究方向，本论文分别针对单列车、多列车以及列车延误后的节能优化运行控制问题开展深入研究。

针对单列车节能优化控制问题，首先分析了单列车双站间驾驶控制策略和列车在不同运行状态下的动力学方程，接着以耗能最小为优化目标，以站间距离、速度限制，站间时间为约束条件分别建立了单列车双站和单列车三站节能运行优化模型，给出了相应的求解算法；利用 MATLAB 软件编程仿真，运用单列车双站节能运行优化模型求解出列车经 A6-A7 站在驾驶控制策略I“牵引-惰行-制动”和策略II“牵引-惰行-巡航-制动”下的最优节能运行速度-距离曲线，以及最优能耗，分别为 $3.457 \times 10^7 J$ 和  $3.872 \times 10^7 J$ ；运用单列车三站节能运行优化模型求解出列车经 A6 – A7 – A8站间的速度距离曲线和最优能耗  $6.824 \times 10^7 J$ 。

针对多列车节能运行优化控制问题，论文综合单列车最优节能和多车协同利用再生能量，实现整体总能耗的降低。对于问题 2-1，将问题一中的三站点运行优化模型进行推广和改进，首先给出了单列车多站点最优节能运行方案，在

此基础上,假设所有列车在轨道线路上具有相同的运行规律和停站时间,以利用的可再生能源最大为优化目标,以相邻列车发车间隔时间为决策变量,建立优化模型,并编程计算获得了单列车在A1 – A2 – … – A14站的运行速度-距离曲线,得到了列车发车的时间间隔  $H=\{h_1,\dots,h_{99}\}$ 。对问题 2-2,考虑早晚高峰期,将一天分成五个时间段,根据不同时间段发车间隔的约束条件,合理分配不同时段的发车数量,将全天的多列车运行问题转化为 5 个多列车节能优化问题,求解得到了五个时间段内列车的发车间隔时间方案。

对于列车延误后运行优化控制问题,首先分析了随机性列车延误时间对列车及运行调整的影响,接着论文建立了基于“延点-赶点结合”调整策略的列车延误后运行优化控制模型,实现尽快恢复正点运行并使能耗增加最小。在模型的求解中,假设某一特定的列车出站延误情景,按照“延点-赶点结合”调整策略对列车运行控制,验证了该调整策略的可行性。

**关键词:** 列车运行优化 节能 再生能源利用 站点延误

## 一、问题重述

轨道交通系统的能耗是指列车牵引、通风空调、电梯、照明、给排水、弱电等设备产生的能耗。根据统计数据,列车牵引能耗占轨道交通系统总能耗 40% 以上。在低碳环保、节能减排日益受到关注的情况下,针对减少列车牵引能耗的列车运行优化控制近年来成为轨道交通领域的重要研究方向。

请研究以下问题:

### 一、单列车节能运行优化控制问题

(1) 建立计算速度距离曲线的数学模型,计算寻找一条列车从  $A_6$  站出发到达  $A_7$  站的最节能运行的速度距离曲线,其中两车站间的运行时间为 110 秒,列车参数和线路参数详见文件“列车参数.xlsx”和“线路参数.xlsx”。

(2) 建立新的计算速度距离曲线的数学模型,计算寻找一条列车从  $A_6$  站出发到达  $A_8$  站的最节能运行的速度距离曲线,其中要求列车在  $A_7$  车站停站 45 秒,  $A_6$  站和  $A_8$  站间总运行时间规定为 220 秒(不包括停站时间),列车参数和线路参数详见文件“列车参数.xlsx”和“线路参数.xlsx”。

### 二、多列车节能运行优化控制问题

(1) 当 100 列列车以间隔  $H=\{h_1, \dots, h_{99}\}$  从  $A_1$  站出发,追踪运行,依次经过  $A_2, A_3, \dots$  到达  $A_{14}$  站,中间在各个车站停站最少  $D_{\min}$  秒,最多  $D_{\max}$  秒。间隔  $H$  各分量的变化范围是  $H_{\min}$  秒至  $H_{\max}$  秒。请建立优化模型并寻找使所有列车运行总能耗最低的间隔  $H$ 。要求第一列列车发车时间和最后一列列车的发车时间之间间隔为  $T_0=63900$  秒,且从  $A_1$  站到  $A_{14}$  站的总运行时间不变,均为 2086s(包括停站时间)。假设所有列车处于同一供电区段,各个车站间线路参数详见文件“列车参数.xlsx”和“线路参数.xlsx”。

补充说明:列车追踪运行时,为保证安全,跟踪列车(后车)速度不能超过限制速度  $V_{limit}$ ,以免后车无法及时制动停车,发生追尾事故。其计算方式可简化如下:

$$V_{limit} = \min(V_{line}, \sqrt{2LB_e})$$

其中  $V_{line}$  是列车当前位置的线路限速 (km/h),  $L$  是当前时刻前后车之间的距离 (m),  $B_e$  是列车制动的最大减速度 ( $m/s^2$ )

(2) 接上问,如果高峰时间(早高峰 7200 秒至 12600 秒,晚高峰 43200 至 50400 秒)发车间隔不大于 2.5 分钟且不小于 2 分钟,其余时间发车间隔不小于 5 分钟,每天 240 列。重新为它们制定运行图和相应的速度距离曲线。

### 三、列车延误后运行优化控制问题

接上问，若列车  $i$  在车站  $A_j$  延误  $DT_j^i$  (10 秒) 发车，建立控制模型，找出在确保安全的前提下，首先使所有后续列车尽快恢复正点运行，其次恢复期间耗能最少的列车运行曲线。

假设  $DT_j^i$  为随机变量，普通延误 ( $0 < DT_j^i < 10s$ ) 概率为 20%，严重延误 ( $DT_j^i > 10s$ ) 概率为 10% (超过 120s, 接近下一班, 不考虑调整), 无延误 ( $DT_j^i = 0$ ) 概率为 70%。若允许列车在各站到、发时间与原时间相比提前不超过 10 秒，根据上述统计数据，研究如何对第二问的控制方案进行调整。

## 二、模型假设

- (1)、列车采用无级调速，列车牵引力和制动力可以在零和最大值之间取连续值；
- (2)、单列车运行时不考虑车长，视为质点；
- (3)、忽略车上乘客的质量，认为列车的质量为一个定值；
- (4)、列车区间运行时间以及列车车站停车时间对所有车次同样适用；

## 三、符号说明

符号	含义	备注
$x$	列车的位移	单位为m
$v$	列车的速度	单位为km/h
$a$	列车的实际加速度	单位为m/s <sup>2</sup>
$M$	列车的重量	单位为kg
$\mu$	实际输出的牵引加速度与最大加速的百分比	
$N$	子区间段数	
$j$	子区段的编号	
$\Delta S_j$	子区段的长度	单位为m
$\Delta T_j$	列车行驶过每个区段所用的时间	单位为 s

$k$	优化运行轨迹的阶段编号	
$v_{jk}$	第 $j$ 子区段优化运行轨迹的第 $k$ 阶段起始速度	
$\Delta t_{jk}$	第 $j$ 子区段优化运行轨迹的第 $k$ 阶段运行时间	单位为 s
$\Delta E_j$	子区间 $j$ 的能耗	单位为 km/h
$g_j$	重力加速度	单位为 $m/s^2$
$\Delta s_{jk}$	子区段 $j$ 中第 $k$ 运行阶段的距离	单位为 m
$E$	总耗能	单位为 J
$D_{min}$	在各个车站列车最小停站时间	单位为 s
$D_{max}$	在各个车站列车最大停站时间	单位为 s
$h_{min}$	最小时间间隔	单位为 s
$E_{reg}$	制动列车产生的再生能量	单位为 J
$E_{used}$	可利用再生能量	单位为 J
$\bar{h}$	平均发车时间间隔	单位为 s
$T_0$	第一列列车发车时间和最后一列列车的发车时间间隔	单位为 s
$S_i$	第 $i$ 个阶段的列车数量	
$DT_j^i$	列车 $i$ 在车站 $A_j$ 延误发车时间	单位为 s

#### 四、问题分析

从轨道交通系统的各方面来看，影响多列车运行过程和列车牵引能耗的因素很多，论文研究面向节能的单/多列车优化决策问题。具体可以分为单列车运行优化控制问题、多列车节能运行优化控制问题和列车延误后运行优化控制问题。

对于问题一，单列车节能优化控制问题。基于单列车双站间驾驶控制策略和列车在牵引、巡航、惰行、制动四种运行状态下的动力学方程，以耗能最小

为最优化目标，以站间距离、速度限制，站间时间为约束条件分别建立单列车双站和单列车三站节能运行优化模型。求解出单列车在双站和三站之间的最节能运行速度距离曲线。

对于问题二，多列车节能运行优化控制问题。对问题 2-1，可综合单列车最优节能和多车协同利用再生能量，实现降低系统总能耗的目标<sup>[1-2]</sup>。将问题一中的三站点运行优化模型进行推广和改进，给出单列车多站点最优节能运行方案。在此基础上，假设所有列车在轨道线路上具有相同的运行规律和停站时间，以利用的可再生能源最大为优化目标，以相邻列车发车间隔时间为决策变量，建立优化模型。对问题 2-2，考虑早晚高峰期，将一天分成五个时间段，根据不同时间段发车时间间隔的约束条件，合理分配不同时段的车数，将全天的多列车运行问题转化为 5 个多列车节能优化问题，然后按照问题 2-1 的方法进行求解。

对于问题三，列车延误后运行优化控制问题。列车出站延误时间具有随机性，首先分析这种延误时间对列车及运行调整的影响，接着建立基于“延点-赶点结合”调整策略的列车延误后运行优化控制模型，实现尽快恢复正点运行并使能耗增加最小。

## 五、问题一：单列车节能运行优化控制问题

### 5.1 模型的准备

#### 5.1.1 单列车双站间驾驶控制策略分析

如果车站间距离较短，列车一般采用“牵引-惰行-制动”的策略运行；

如果站间距离较长，列车通常会采用牵引到接近限制速度后，交替使用惰行、巡航、牵引三种工况，直至接近下一车站采用制动进站停车。

这里考虑到城市轨道交通多限速、多坡度、站间线路距离不等等复杂性，研究采用“牵引-惰行-巡航-制动”和“牵引-惰行-制动”两种驾驶控制策略下的单列车节能运行优化模型。

### 5.1.2 列车不同运行状态的动力学方程

在建立列车定时节能优化模型之前，需要了解列车四种运行状态下的动力学关系。所以论文首先根据牛顿第二定律求得列车不同运行状态下的动力学方程<sup>[2-3]</sup>。

(1) 牵引（加速）状态

$$\begin{cases} v = \frac{dx}{dt} * 3.6 \\ a = \frac{dv}{dt} / 3.6 = \frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000} \end{cases}$$

(2) 巡航状态

$$\begin{cases} v = \text{const} \\ a = \frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000} = 0 \end{cases}$$

(3) 惰行状态

$$\begin{cases} v = \frac{dx}{dt} * 3.6 \\ a = \frac{dv}{dt} / 3.6 = -\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000} \end{cases}$$

(4) 制动状态

$$\begin{cases} v = \frac{dx}{dt} * 3.6 \\ a = \frac{dv}{dt} / 3.6 = \frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000} \end{cases}$$

其中,  $x$  为列车的位移, 单位为  $\text{m}$ ,  $v$  为列车在位置  $x$  处的速度, 单位为  $\text{km/h}$ ,  $a$  为加速度, 单位为  $\text{m/s}$ ,  $M$  为列车的重量, 单位为  $\text{kg}$ 。  $\mu_1$  为实际输出的牵引加速度与最大加速的的百分比,  $\mu_2$  为实际输出的制动加速度与最大加速的的百分比。

## 5.2 单列车双站间定时节能运行优化模型的建立

(1) 区段划分和驾驶阶段序列选择<sup>[4]</sup>

在将曲率因素折算为坡度因素后，首先将线路按限速和坡度分段成很多子区段。经过分段后定义最终的子区间段数为  $N$ , 每个子区间的编号为  $j$ ,  $j \in \mathbb{Z}$ ,  $1 \leq j \leq N$ , 每个子区段的长度为  $\Delta S_j$ , 列车行驶过每个区段所用的时间为  $\Delta T_j$ 。

策略I：列车按照“牵引-惰行-制动”运行；



策略II：列车按照“牵引-巡航-惰行-制动”运行。

策略I情况下，将状态按顺序分为3个阶段，并将优化运行轨迹的阶段编号为 $k = 1, 2, 3$ 。

策略II情况下，将状态按顺序分为4个阶段，并将优化运行轨迹的阶段编号为 $k = 1, 2, 3, 4$ 。

在研究单个子区段时，应该同时考虑驾驶控制序列和前、后子区段的限速。定义每个阶段的起始速度为 $v_{jk}$ 其中 $j$ 为子区段的编号, $k$ 为优化运行轨迹的阶段编号，列车在这个子区段的出口速度为 $v_{(j+1)1}$ 。

每个阶段的运行时间为 $\Delta t_{jk}$ 。

## (2) 列车的能耗

由前面的分析知，当按照策略I运行时，列车只在牵引加速状态下耗能。当按照策略II运行时，列车牵引加速和巡航状态下耗能。

记 $\Delta E_j$ 为子区间 $j$ 的能耗，

策略I情况下：

$$\Delta E_j = \int_{v_{j1}}^{v_{j2}} \frac{\mu_1 f_F(v) v / 3.6}{\frac{1000 \mu_1 f_F(v)}{M} - \left( w_0 + w_1 + i + \frac{c}{R} \right) * \frac{g}{1000}} dv$$

策略II情况下：

$$\Delta E_j = \int_{v_{j1}}^{v_{j2}} \frac{\mu_1 f_F(v) v / 3.6}{\frac{1000 \mu_1 f_F(v)}{M} - \left( w_0 + w_1 + i + \frac{c}{R} \right) * \frac{g}{1000}} dv + \Delta t_{j2} \cdot v_{j2} \mu_1 f_F(v)$$

## (3) 优化目标

优化目标是减小列车的运行能耗，即：

$$\min \sum_{j=1}^N \Delta E_j$$

## (4) 约束条件

约束条件包括限制速度约束、时间约束、距离约束。

1) 限制速度约束：列车的运行速度不能超过轨道限速,即：

$$0 \leq v_{jk} \leq v_{lim}^j, \quad 1 \leq j \leq N,$$

$$k = \begin{cases} 1,2,3. (\text{策略 I}) \\ 1,2,3,4 (\text{策略 II}) \end{cases}$$

2) 时间约束: 列车在各个子区段上运行的时间之和应等于总的运行时间。  
所以有:

$$\sum_{j=1}^N \sum_{k=1}^3 \Delta t_{jk} = T \quad (\text{策略 I})$$

$$\sum_{j=1}^N \sum_{k=1}^4 \Delta t_{jk} = T \quad (\text{策略 II})$$

同时,子区段 $j$ 中各个运行阶段的时间 $\Delta t_{jk}$ 应该不小于 0。

$$\Delta t_{jk} \geq 0$$

3) 距离约束: 列车在各个子区段的运行距离之和应与站间距离相等。

$$\sum_{j=1}^N \sum_{k=1}^3 \Delta s_{jk} = L \quad (\text{策略 I})$$

$$\sum_{j=1}^N \sum_{k=1}^4 \Delta s_{jk} = L \quad (\text{策略 II})$$

对于每个子区段, 第一个运行阶段有可能是牵引加速状态, 运行时间和距离可以通过 $v_{j1}$ 与 $v_{j2}$ 表示, 可得:

$$\Delta t_{j1} = \int_{v_{j1}}^{v_{j2}} \frac{1/3.6}{\frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

$$\Delta s_{j1} = \int_{v_{j1}}^{v_{j2}} \frac{v/3.6}{\frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

策略I情况下:

第二个阶段是惰行状态, 运行时间和距离可以通过 $v_{j2}$ 与 $v_{j3}$ 表示:

$$\Delta t_{j2} = \int_{v_{j2}}^{v_{j3}} \frac{1/3.6}{-\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

$$\Delta s_{j2} = \int_{v_{j2}}^{v_{j3}} \frac{v/3.6}{-\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

第三个阶段是制动减速状态, 运行时间和距离可通过 $v_{j3}$ 和 $v_{(j+1)1}$ 表示:

$$\Delta t_{j3} = \int_{v_{j3}}^{v_{(j+1)1}} \frac{1/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

$$\Delta s_{j3} = \int_{v_{j3}}^{v_{(j+1)1}} \frac{v/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

根据优化目标和约束条件，可得策略I情况下列车的节能优化模型：

$$\min \sum_{j=1}^N \Delta E_j \quad \text{s.t.} \left\{ \begin{array}{l} 0 \leq v_{jk} \leq v_{lim}^j \\ \sum_{j=1}^N \sum_{k=1}^3 \Delta t_{jk} = T \\ \sum_{j=1}^N \sum_{k=1}^3 \Delta s_{jk} = L \\ \Delta t_{jk} \geq 0 \\ \Delta t_{j1} = \int_{v_{j1}}^{v_{j2}} \frac{1/3.6}{\frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ \Delta s_{j1} = \int_{v_{j1}}^{v_{j2}} \frac{v/3.6}{\frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ \Delta t_{j2} = \int_{v_{j2}}^{v_{j3}} \frac{1/3.6}{-\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ \Delta s_{j2} = \int_{v_{j2}}^{v_{j3}} \frac{v/3.6}{-\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ \Delta t_{j3} = \int_{v_{j3}}^{v_{(j+1)1}} \frac{1/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ \Delta s_{j3} = \int_{v_{j3}}^{v_{(j+1)1}} \frac{v/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ 1 \leq j \leq N \\ k = 1, 2, 3 \end{array} \right.$$

策略II情况下：

第二个运行阶段是巡航状态，运行时间 $\Delta t_{j2}$ 作为决策变量，距离可以表示为：

$$\Delta s_{j2} = \Delta t_{j2} \cdot v_{j2}$$

第三个阶段是惰行状态，运行时间和距离可以通过 $v_{j3}$ 与 $v_{j4}$ 表示：

$$\Delta t_{j3} = \int_{v_{j3}}^{v_{j4}} \frac{1/3.6}{-\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

$$\Delta s_{j3} = \int_{v_{j3}}^{v_{j4}} \frac{v/3.6}{-\left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

第四个阶段是制动减速状态，运行时间和距离可通过 $v_{j4}$ 和 $v_{(j+1)1}$ 表示：

$$\Delta t_{j4} = \int_{v_{j4}}^{v_{(j+1)1}} \frac{1/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

$$\Delta s_{j4} = \int_{v_{j4}}^{v_{(j+1)1}} \frac{v/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv$$

根据优化目标和约束条件，可得列车的节能优化模型：

$$\begin{aligned} & \min \sum_{j=1}^N \Delta E_j \\ & \text{s.t.} \left\{ \begin{aligned} & 0 \leq v_{jk} \leq v_{lim}^j \\ & \sum_{j=1}^N \sum_{k=1}^4 \Delta t_{jk} = T \\ & \sum_{j=1}^N \sum_{k=1}^4 \Delta s_{jk} = L \\ & \Delta t_{jk} \geq 0 \\ & \Delta t_{j1} = \int_{v_{j1}}^{v_{j2}} \frac{1/3.6}{\frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ & \Delta s_{j1} = \int_{v_{j1}}^{v_{j2}} \frac{v/3.6}{\frac{1000\mu_1 f_F(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ & \Delta s_{j2} = \Delta t_{j2} \cdot v_{j2} \\ & \Delta t_{j3} = \int_{v_{j3}}^{v_{j4}} \frac{1/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ & \Delta s_{j3} = \int_{v_{j3}}^{v_{j4}} \frac{v/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ & \Delta t_{j4} = \int_{v_{j4}}^{v_{(j+1)1}} \frac{1/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ & \Delta s_{j4} = \int_{v_{j4}}^{v_{(j+1)1}} \frac{v/3.6}{\frac{-1000\mu_2 f_B(v)}{M} - \left(w_0 + w_1 + i + \frac{c}{R}\right) * \frac{g}{1000}} dv \\ & 1 \leq j \leq N, k = 1, 2, 3, 4 \end{aligned} \right. \end{aligned}$$

### 5.3 单列车双站节能优化模型求解与分析

#### 5.3.1 模型求解设计

从上一小节中可以看出，单车节能优化控制模型是一个同时含有等式约束和不等式约束的非线性规划问题，其实质是求解最小能耗下的 $\mu_1$ ， $\mu_2$ ， $\Delta t_{jk}$ 。为求解这一问题，论文考虑将总的运行时间 $T$ 按等时间间隔 $1s$ 离散化 $t = 1, 2, \dots, T$ ，<sup>[5-7]</sup>在两站之间列车按照“牵引-惰行-制动”的驾驶控制策略运行。结合运行特点和动力学模型，时间离散化后的列车运行过程如下：

(1) 离站启动：结合实际，合理设定 $\mu_1$ ， $\mu_1$ ，牵引加速时间 $t_1$ ，制动时间 $t_2$ 循环搜索范围， $v(0) = 0, s(0) = 0$ ，列车启动；

(2) 轨道情况识别：分析轨道坡度  $i$ ，曲率  $R$ ，限速 $V_{limit}$ ；

(3) 自身动力学状态判别：观察时间  $t$ ， $v(t)$ ， $s(t)$ 。判别  $t$  与牵引加速时间 $t_1$ 、 $t_2$ 、 $T$ 的关系确定运行状态，判别  $v(t)$ 与 $V_{limit}$ 的大小， $s(t)$ 与  $L$  的大小；

(4) 动力学参数及耗能计算：

**Step1:** 由 $v(t)$ 求解牵引力 $F(t) = \mu_1 f_F(v(t))$ ，根据牛顿第二定律，求解动力学参数 $a(t)$ ；

**Step2:** 认为  $t-(t+1)$  这  $1s$  的时间间隔内列车近似以 $a(t)$ 做匀加速运动  
 $v(t+1) = v(t) + a(t)$  ；

**Step3:**  $\Delta s(t) = [v(t+1) + v(t)]/2$ ， $s(t+1) = s(t) + \Delta s(t)$ ；

**Step4:**  $t-(t+1)$  这  $1s$  的时间间隔内列车牵引力耗能 $\Delta E(t) = F(t) \cdot \Delta s(t)$   
 $E(t+1) = E(t) + \Delta E(t)$

(5) 到站停止： $t=T, s(t) = L, v(t) = 0$ ；

(6) 对设定的每一组 $(\mu_1, \mu_2, t_1, t_2)$ 求解记录总耗能 $E(t_1)$ ，并找到其中的最大值。对应的运行方案即为搜索确定的优化节能运行方案。

### 5.3.2 A6-A7 双站列车运行优化结果及分析

首先，建立 A6-A7 站间单列车定时节能控制模型。其中 $T = 110s$ ， $L = 1354m$ ，A6-A7 站间线路曲率为 0，站间线路的坡度参数如表 5-1 所示。

表 5-1 A6-A7 站之间的线路坡度参数表

坡道位置/m	0-304	304-684	684-1304	1304-1354
坡度(‰)	0	3.5	-1.8	0

将运行时间离散化后，仿真求解策略I和策略II情况下的速度-距离、速度-时间、牵引力-距离曲线和能耗曲线。A6-A7 站间列车运行速度距离曲线的每一秒数据见附件。

(1) 策略I 和策略II驾驶控制方式下的列车节能运行优化结果

1)、策略I：列车按照“牵引-惰行-制动”驾驶优化结果

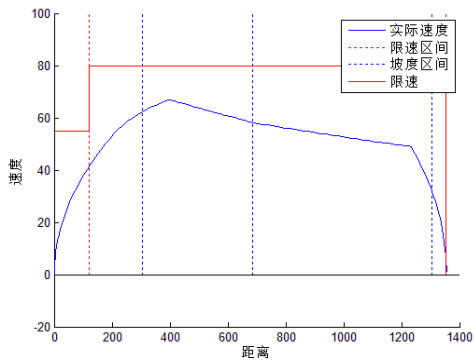


图 5-1 策略I 运行 v-s图

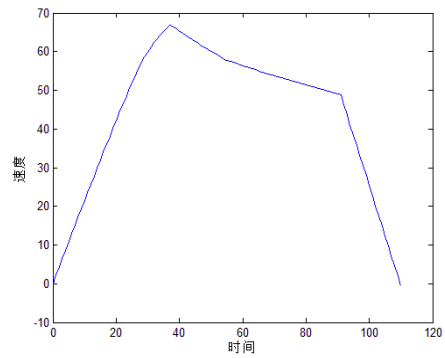


图 5-2 策略I 运行 v-t 图

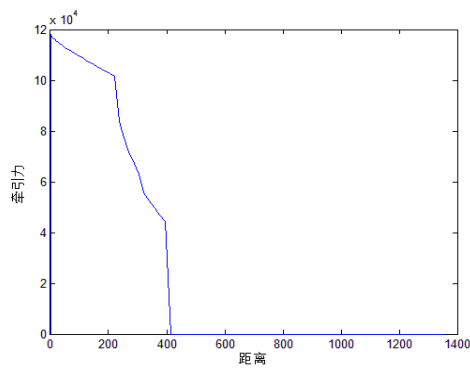


图 5-3 策略I牵引力-距离曲线

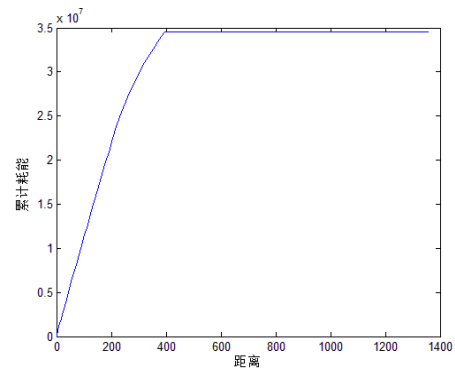


图 5-4 策略I列车能耗曲线

2)、策略II：列车按照“牵引-巡航-惰行-制动” 驾驶优化结果

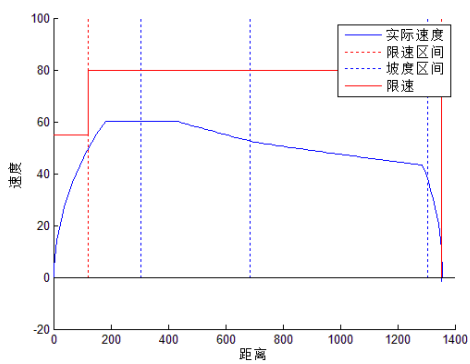


图 5-5 策略II 运行 v-s图

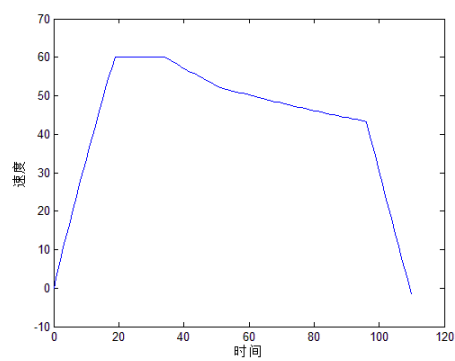


图 5-6 策略II 运行 v-t 图

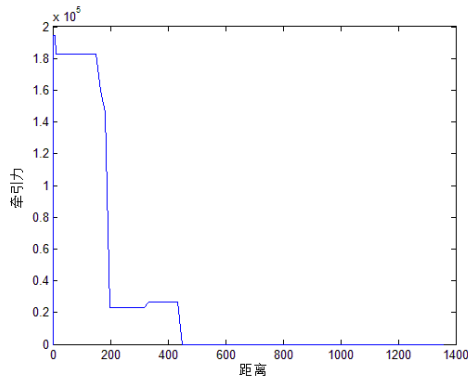


图 5-7 策略II牵引力-距离曲线

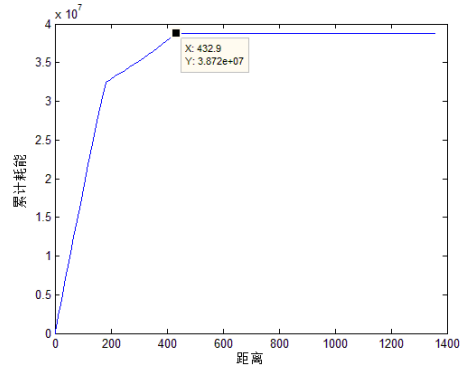


图 5-8 策略II列车能耗曲线

## (2) 结果分析

1)、由列车运行的 $v-s$ 图，可以看出两种策略下均能实现节能优化，但优化结果不同，策略I的最终优化能耗为 $3.457 \times 10^7 \text{J}$ ,策略II的最终优化能耗为 $3.872 \times 10^7 \text{J}$ ;

2)、运动情况对比如表 5-2 所示

表 5-2 策略I和策略II运动情况对比

运动参数	策略I	策略II
牵引加速时间 s	37	19
最大速度 km/h	66.95	60.04
制动减速时间 s	18	13
最大制动加速度 $\text{m/s}^2$	-0.74	-0.89

由上表可知，在 A6-A7 站之间采用策略I牵引加速的时间，最大速度比、制动减速时间策略II大，最大制动加速度比策略II小；

## 5.4 单列车三站定时节能运行优化模型的建立

针对问题一、第二小问 A6 站-A7 站-A8 站列车运行要求，首先建立单列车三站优化模型。该单列车三站优化问题与两站的区别是：三站总的运行时间一定，在中间站点的停车时间一定，而在两段单独运行的时间不确定。所以在两站一段模型的基础上，需要考虑三站两段运行中的时间分配问题。

这里，论文需要增加决策变量 $T_{1,2}$ ， $T_{2,3}$ 。 $T_{1,2}$ 代表列车在站点 1 和站点 2 之间运行时间， $T_{2,3}$ 代表列车在站点 2 和站点 3 之间运行的时间。且满足附加约束条件：

$$T_{1,2} + T_{2,3} = T$$

三站问题首先也要对每两站之间根据线路坡道和限速状况进行区段划分。分别划分成 $N_1$ 个子区段, $j = 1, 2, \dots, N_1$ 和 $N_2$ 个子区段 $l = 1, 2, \dots, N_2$ 。同上一节的分析，可以得到三站优化模型如下：

$$\min \sum_{j=1}^{N_1} \Delta E_j + \sum_{l=1}^{N_2} \Delta E_l$$

其他约束条件同前面模型，这里不再赘述。

## 5.5 单列车三站节能运行优化模型求解与分析

### 5.5.1 模型求解设计

5.3 节已经对双站间的节能优化求解进行了仿真设计，并实际应用到 A6-A7 站的列车运行优化情况，求解效果较好。但对于三站问题，增加了决策变量 $T_{1,2}$ ，即增加了在两站之间运行时间的分配问题，使得问题求解的规模变大。我们通过设计更加合理的站间运行时间分配方法实现减小搜索范围快速求解。每两站之间的时间分配确定后，对于两站间的优化，论文继续沿用上面模型的求解方法。

一般来说，城市轨道交通中两站之间线路越长，列车运行的时间也越长。所以，在分配 $T_{12}$ 和 $T_{23}$ 时，初值设定结合线路长度情况。

由题意可知，列车运行时间和能耗近似满足反比关系。如图 5-9 所示。

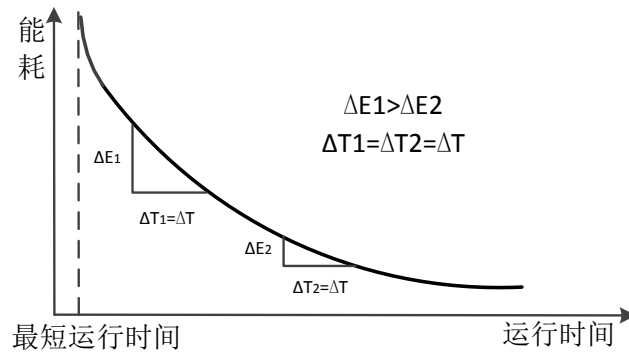


图 5-9 站间运行与能耗关系



由关系图可知：

(1) 在固定的运行时间，同段轨道线路具有最优能耗；

(2) 对同段轨道线路，增加相同的运行时间不一定会减少等量的能耗。列车站间运行时间与能耗变化的趋势影响能耗的减少。

所以，我们可以合理外推：对于不同的轨道线路均具有对应的最优能耗-运行时间关系，并且增加相同的运行时间也不一定减少等量的能耗。据此我们可以得到时间调整的策略。

结合上面的分析，论文设计如下调整  $(T_{1,2}, T_{2,3})$  取值不断寻求最优解的方法。

Step1: 结合线路长度情况预设  $(T_{1,2}^0, T_{2,3}^0)$ ，并且满足：

$$\begin{cases} T_{1,2}^0 + T_{2,3}^0 = T \\ \frac{T_{1,2}^0}{T_{2,3}^0} = \frac{L_{1,2}}{L_{2,3}} \end{cases}$$

Step2: 求解出列车在  $(T_{1,2}^0, T_{2,3}^0)$  条件下的优化解  $(E_{1,2}^0, E_{2,3}^0)$ ，总耗能  $E^0 = E_{1,2}^0 + E_{2,3}^0$ ；

Step3: 时间分配改变  $\Delta T_1$ 。分配方案为  $(T_{1,2}^1, T_{2,3}^1)$ ，其中：

$$\begin{cases} T_{1,2}^1 = T_{1,2}^0 + \Delta T_1 \\ T_{2,3}^1 = T_{2,3}^0 - \Delta T_1 \end{cases}$$

Step4: 求解出列车在  $(T_{1,2}^1, T_{2,3}^1)$  条件下的优化解  $(E_{1,2}^1, E_{2,3}^1)$ 、能耗改变  $(\Delta E_{1,2}^1, \Delta E_{2,3}^1)$ 、总耗能  $E^1 = E_{1,2}^1 + E_{2,3}^1$ 、 $\Delta E^1$ 。

Step5: 能耗改变判别及调整策略

(1) 若  $\Delta E^1 < 0$ ，由能耗-运行时间关系可知，此时  $\Delta E_{1,2}^1 < 0$ ， $\Delta E_{2,3}^1 > 0$ ， $|\Delta E_{1,2}^1| > |\Delta E_{2,3}^1|$ ，说明这个方向的时间分配改变调整对降低能耗是有效的，下一次迭代取与  $\Delta T_1$  相同的符号，取  $\Delta T_1 = 0.5\Delta T_1$ ，返回 Step3。

(2) 若  $\Delta E^1 > 0$ ，有两种可能性，需要进一步求解判定调整策略。

1)、取  $\Delta T_1 = -\Delta T_1$ ，返回 Step3，有  $\Delta E^1 < 0$ ，则令  $\Delta T_1 = 0.5\Delta T_1$ ，返回 Step3。

2)、取  $\Delta T_1 = -\Delta T_1$ ，返回 Step3，有  $\Delta E^1 > 0$ ，则再令  $\Delta T_1 = -0.5\Delta T_1$ ，返回 Step3。

Step6: 当按照上面的步骤不断迭代使 $\Delta T_1 < 1s$  时, 停止调整, 此时的 $(T_{1,2}^1, T_{2,3}^1)$  即可以认为是最佳的时间分配方案, 对应的可以求出最优的列车运行方案和总耗能 $E^1$ 。

### 5.5.2 A6-A7-A8 三站列车运行优化结果及分析

根据线路长度情况预分配  $(T_{6,7}^0, T_{7,8}^0) = (112, 108)$ 秒

运用 MATLAB 软件仿真求解得到相对较优解:  $T_{6,7} = 106s, T_{7,8} = 114s$ 。并求解计算得到在时间分配  $(T_{6,7}, T_{7,8}) = (106, 114)$  秒情况下的最优能耗为  $(E_{6,7}, E_{7,8}) = (3.679, 3.145) \times 10^7 J$ , 总能耗  $E = 6.824 \times 10^7 J$ 。列车在 A6-A7-A8 站间运行曲线如图 5-10, 图 5-11, 牵引力-距离曲线、列车能耗曲线分别如图 5-12 和图 5-13。A6-A7 站间列车运行速度距离曲线的每一秒数据见附件。

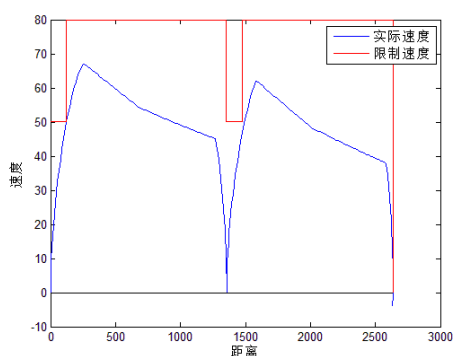


图 5-10 A6-A7-A8 节能运行 v-s 图

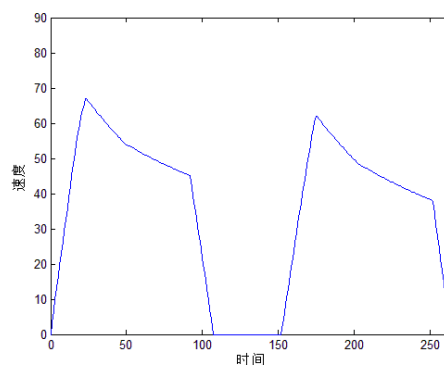


图 5-11 A6-A7-A8 节能运行 v-t 图

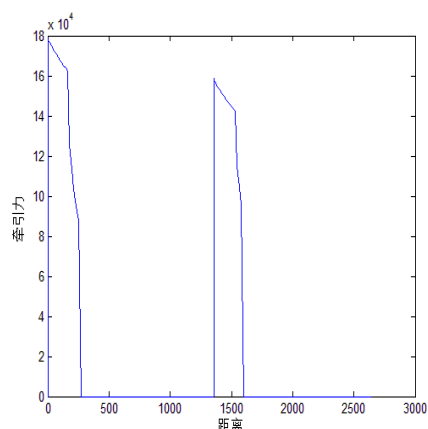


图 5-12 A6-A7-A8 运行牵引力-距离曲线

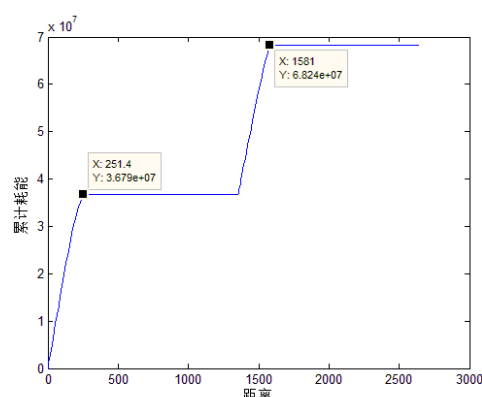


图 5-13 A6-A7-A8 列车能耗曲线

## 六、问题二：多列车节能运行优化控制问题

### 6.1 单列车多站节能运行优化模型

#### 6.1.1 模型的建立

将问题一中的单列车三站定时节能运行优化模型进行推广和改进，可以得到多站优化模型。

设有  $N$  个车站，在各个车站列车停站时间最少  $D_{min}$  秒，最大  $D_{max}$  秒，由能耗-运行时间关系图，我们知道增加列车在站间运行时间可以减小耗能。因此我们假设列车在各个车站的停车时间均为最小值  $D_{min}$  秒。以列车在各个站间运行的时间为决策变量，分别为  $T_{m,n}$ ，这里  $m, n$  为车站序号， $n = m + 1$ ，

$T_{m,n} = T_{1,2}, T_{2,3}, T_{3,4}, T_{4,5}, \dots, T_{13,14}$ ，并且满足：

$$\sum T_{mn} + (N - 2) * D_{min} = T$$

其中， $T$  为指定的列车从起始站点出发到最后一个站点的总时间。

和三站问题相同的是，单列车多站优化问题也要对每两站之间根据线路坡道和限速状况进行区段划分。分别划分成  $N_1$ 、 $N_2$ 、 $N_3$ 、...、 $N_{N-1}$  个子区段。

同问题一的分析，可以得到单列车多站优化模型如下：

$$\min \sum_{j=1}^{N_1} \Delta E_j + \sum_{j=2}^{N_2} \Delta E_{j2} + \sum_{j=3}^{N_3} \Delta E_{j3} + \dots + \sum_{j(N-1)=1}^{N_{(N-1)}} \Delta E_{j(N-1)}$$

其他约束条件同单列车双站基本模型。

#### 6.1.2 模型的求解

##### (1) 求解策略

问题二列车经过 A1-A2-A3-...-A14 站，同 5.5 节中三站节能运行优化模型的求解，论文结合线路情况和能耗-运行时间关系设计比较合理的各站时间分配方案，转化为 13 个单车双站问题求解。

**Step1:** 结合线路长度情况预设定  $(T_{1,2}^0, T_{2,3}^0, T_{3,4}^0, T_{4,5}^0, \dots, T_{13,14}^0)$ ，并且满足：

$$\begin{cases} \sum T_{m,n}^0 = T - (N - 2) * D_{min} \\ \frac{T_{1,2}^0}{T_{2,3}^0} = \frac{L_{1,2}}{L_{2,3}}, \frac{T_{2,3}^0}{T_{3,4}^0} = \frac{L_{2,3}}{L_{3,4}}, \dots \end{cases}$$

设定优化调整时间分配的次数 I;

Step2: 求解出列车在  $(T_{1,2}^0, T_{2,3}^0, T_{3,4}^0, T_{4,5}^0, \dots, T_{13,14}^0)$  条件下的优化解

$(E_{1,2}^0, E_{2,3}^0, \dots, E_{13,14}^0)$ , 总耗能  $E_i^0 = \sum E_{m,n}^0$ ,  $E_i^0$  代表第 i 次求解出的耗能;

Step3: 设定  $\Delta T = 1s$ 。在原来时间分配  $(T_{1,2}^0, T_{2,3}^0, T_{3,4}^0, \dots, T_{13,14}^0)$  的基础上分别增加  $\Delta T$ , 求解出  $(E_{1,2}^1, E_{2,3}^1, \dots, E_{13,14}^1)$  和  $(\Delta E_{1,2}^1, \Delta E_{2,3}^1, \dots, \Delta E_{13,14}^1)$ , 并将  $(\Delta E_{1,2}^1, \Delta E_{2,3}^1, \dots, \Delta E_{13,14}^1)$  按照大小进行排序;

Step4: 根据能耗变化情况, 将能耗增加排在后六位的站间时间分到能耗增加排在前六位的站间, 新的时间分配为  $(T_{1,2}^1, T_{2,3}^1, T_{3,4}^1, \dots, T_{13,14}^1)$ , 令  $i=i+1$ ,

判定: 当  $i>I$  时停止计算。否则令  $(T_{1,2}^0, T_{2,3}^0, T_{3,4}^0, \dots, T_{13,14}^0) =$

$(T_{1,2}^1, T_{2,3}^1, T_{3,4}^1, \dots, T_{13,14}^1)$ , 返回到 Step2;

Step5: 确定的相对较优的耗能为  $E_{10}^0$ , 时间分配为停止计算时的

$(T_{1,2}^1, T_{2,3}^1, T_{3,4}^1, \dots, T_{13,14}^1)$ , 各站间运行情况也相应得到。

## (2) 模型的结果与分析

按照上面的求解策略, 利用 MATLAB 软件编程求解。得到:

1)、 $(T_{1,2}^0, T_{2,3}^0, T_{3,4}^0, T_{4,5}^0, \dots, T_{13,14}^0)$

$= (95, 101, 161, 170, 169, 96, 95, 115, 79, 143, 177, 100, 196)$  秒。

调整后相对较优的时间分配集合为:

$(T_{1,2}, T_{2,3}, T_{3,4}, T_{4,5}, \dots, T_{13,14})$

$= (90, 106, 166, 171, 164, 91, 95, 115, 78, 138, 177, 105, 196)$  秒。

2)、相对较优的能耗集合为:

$(E_{1,2}, E_{2,3}, \dots, E_{13,14})$

$= (4.46, 4.39, 4.39, 4.62, 4.99, 4.09, 4.12, 3.93, 4.26, 4.63, 4.95, 4.37, 5.34) \times 10^7 J$

3)、总耗能 $E = 58.52 \times 10^7 \text{J}$ 。

4)、单个列车在 A1-A2-A3-...-A14 站的速度-距离曲线如图 6-1 所示。

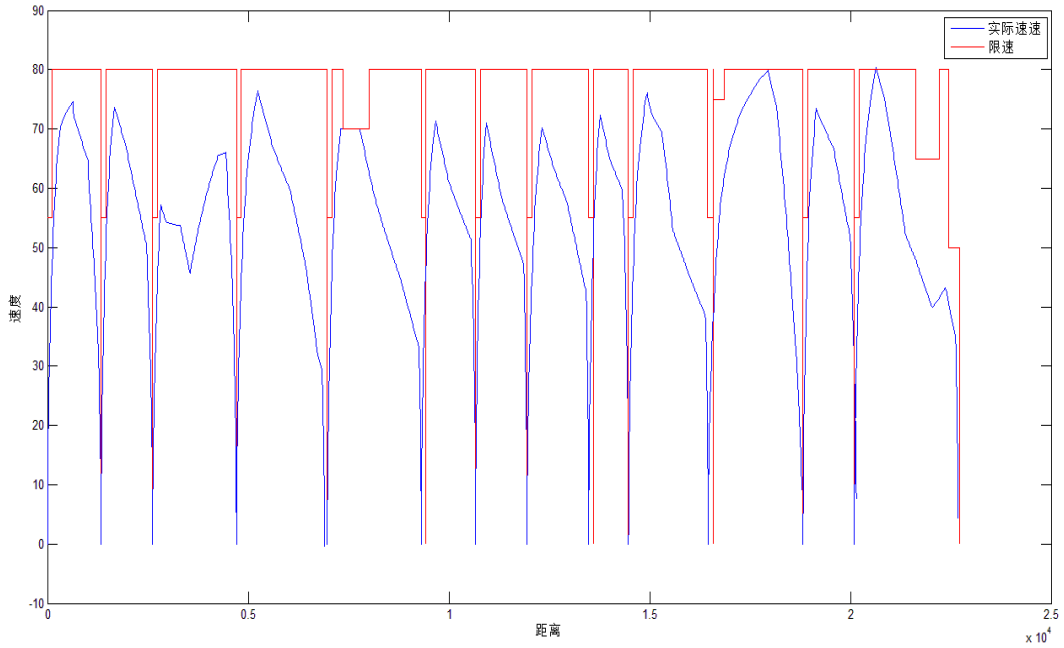


图 6-1 单列车 A1-A2-A3-...A14 站节能运行速度-距离曲线

## 6.2 再生能量利用下多列车节能运行优化模型

### 6.2.1 模型的建立

基于所有列车在相同的站间运行完全一样，所有车站停站时间相同的假设，以发车时间间隔  $H=\{h_1, \dots, h_{99}\}$  为决策变量，建立再生能量利用的运行优化模型。

#### (1) 约束条件

在合理假设将问题简化后，只考虑发车间隔时间约束和总间隔时间约束：

①发车间隔时间约束：

$$h_{min} < h_i < h_{max}, \quad i = 1, 2, 3, \dots, 99$$

②总间隔时间约束：

$$\sum_{i=1}^{99} h_i = T_0$$

$T_0$ 为总间隔时间，且 $T_0=63900\text{s}$ 。

#### (2) 目标函数

论文以所有列车再生能量利用量最高为目标函数。目标函数是建立在对列车运行过程分析的基础上。

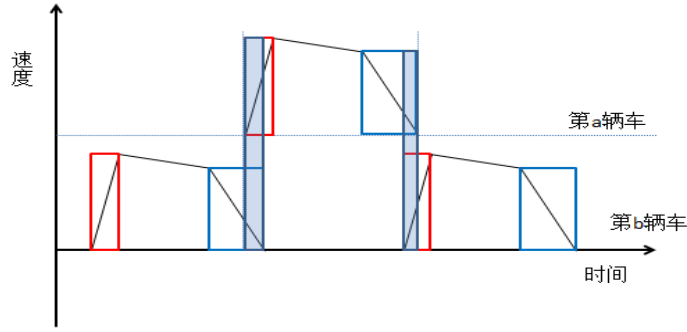


图 6-2 再生能量利用原理示意图

再生能量利用原理如图 6-2 所示。列车 a 在制动时会产生能量 $E_{reg}$ ，如果轨道上还有列车 b 处于加速状态，其可以利用 $E_{reg}$ ，从而减少从变电站获得的能量，达到节能的目的。如果列车 a 制动时，其所处供电区段内没有其他列车加速，其产生的再生能量不能被转化为牵引用能量。

制动列车产生的再生能量

$$E_{reg} = (E_{mech} - E_f) \cdot 95\%$$

其中 $E_{mech}$ 是制动过程中列车机械能的变化量， $E_f$ 是制动过程中为克服基本阻力和附加阻力所做功。

被利用了的再生能量可按照以下假设的公式计算

$$E_{used} = E_{reg} \cdot t_{overlap} / t_{brake}$$

其中 $t_{overlap}$ 是列车 a 制动的时间与列车 b 加速时间的重叠时间， $t_{brake}$ 是列车 i 的制动时间。即制动时所产生的再生能量与制动时间成正比。在一定发车间时间间隔  $H=\{h_1, \dots, h_{99}\}$  下，所有列车的运行均确定，

记第 i 辆列车第 j 次制动产生的被利用的再生能量为 $E_{used(ij)}$ ，则总的被利用的再生能量为 $\sum_{i=1}^{100} \sum_{j=1}^{13} E_{used(ij)}$ 。

所以优化目标为:

$$\max \sum_{i=1}^{100} \sum_{j=1}^{13} E_{used(ij)}$$

### 6.2.2 模型的求解

考虑再生能量利用的多列车运行优化模型的求解非常复杂，很难求得全局的最优解。在求解时，我们结合实际对模型进行简化处理，实现尽量增加再生能量利用，减小能耗的目的。

#### (1) 求解策略

为使问题简化我们假设制动列车在不同的制动过程中产生的再生能量 $E_{reg}$ 相同。我们求取 13 个站间制动过程产生的再生能量值，取它们的平均值作为再生能量 $E_{reg}$ 。

这样处理后问题转化为以列车运行中的牵引-制动重叠时间最大化为目标函数。重叠时间是指列车之间的牵引过程与制动过程的时间重叠。

根据题目要求第一列列车发车时间和最后一列列车的发车时间间隔为 $T_0=63900$  秒，平均发车时间间隔为 $\bar{h}=645.45$  秒。 $\bar{h}$ 取整为 645 秒。

假设列车发车的间隔时间都在平均值附近，这种假设基于生活实际。一般来说城市轨道交通为了乘客方便，采取规律地接近均匀地发车。从 A1 站到 A14 站的总运行时间不变，均为 2086s（包括停站时间），则在整条轨道线路上最多同时运行 4 辆车。

考虑前面出发的 5 辆车，第 2 辆车在第 1 辆车出发 $\bar{h}$ 秒附近且第一辆车制动的过程中出发，按照这样的策略确定 $h_1$ ，保证了再生能量的利用；以后的列车按照同样的思想处理，第 3 辆车在第 1 辆车出发  $2\bar{h}$ 秒附近且第 1 辆车或第 2 辆车制动开始时出发，确定 $h_2$ ；第 4 辆车在第 1 辆车出发  $3\bar{h}$ 秒附近且第 1 辆车、第 2 辆车或第 3 辆车制动开始时出发，确定 $h_3$ ；第 5 辆车出发时第 1 辆车已经到达 A14 站，所以第 5 辆车在第 4 辆车出发 $\bar{h}$ 秒附近且第 2、3、4 车制动开始时出发，确定 $h_4$ 。以后的发车策略同前面 5 辆车，不断确定 $h_5, h_6, \dots, h_{98}$ 。最后一辆车按要求在第 1 辆车出发  $T_0=63900$ s 后出发，这样确定 $h_{99}$ 。

在确定再生能量牵引-制动重叠时间时，我们按照确定发车时间间隔

$(h_1, h_2, h_3, h_4)$  的思路，每 5 辆车一组进行分析计算它们的牵引-制动重叠时间比ratio。依次类推，分析计算所有的牵引-制动重叠时间比ratio。总的再生能量利用为所有的再生能量利用之和，即：

$$\sum E_{used} = \sum \text{ratio} \cdot E_{reg}$$

## (2)模型的结果与分析

按照上面的思路进行分析求解，得到列车的发车间隔时间安排如表 6-1 所示。

表 6-1 列车发车间隔时间安排

车次	1	2	3	4	5	6	7	8	9	10
间隔		656	647	641	635	655	631	654	650	656
车次	11	12	13	14	15	16	17	18	19	20
间隔	644	634	642	640	632	634	649	646	640	658
车次	21	22	23	24	25	26	27	28	29	30
间隔	630	646	657	656	653	645	652	641	653	630
车次	31	32	33	34	35	36	37	38	39	40
间隔	650	645	643	644	658	648	638	652	639	638
车次	41	42	43	44	45	46	47	48	49	50
间隔	659	651	643	639	652	659	652	653	645	635
车次	51	52	53	54	55	56	57	58	59	60
间隔	638	641	647	631	638	659	643	634	651	657
车次	61	62	63	64	65	66	67	68	69	70
间隔	644	651	657	638	652	637	643	652	653	634
车次	71	72	73	74	75	76	77	78	79	80
间隔	644	656	638	644	650	630	641	634	643	634
车次	81	82	83	84	85	86	87	88	89	90
间隔	655	657	655	633	644	654	652	656	651	646
车次	91	92	93	94	95	96	97	98	99	100
间隔	643	648	632	635	657	647	640	645	635	651

## 6.3 考虑高峰期的多列车运行节能优化模型

每天的不同时段，在城市轨道交通运行过程中，存在客流的早晚高峰期。为了更好地发挥列车的运能，应合理调度列车，优化不同时期的发车间隔。



图 6-3 一天中的客流规律



一天中的客流按时间可以分成如图 6-3 所示的五个阶段：早高峰前，早高峰，早高峰与晚高峰之间，晚高峰，晚高峰后。

考虑高峰期的多列车运行节能问题可以描述为：一定的列车数量 $S_{\text{车}}=240$ ，在高峰期和非高峰期发车时间间隔要求不同，结合高峰期和非高峰期的不同要求，确定合理的列车运行方案。论文在上一节模型的基础上进行改进，增加一些决策变量，建立考虑高峰期的多列车运行节能优化模型。

### 6.3.1 模型的建立

(1)约束条件：

以早高峰、晚高峰为节点可以分成五个阶段的列车，定为 1, 2, 3, 4, 5 类车，分别是：早高峰前列车，早高峰列车，早高峰与晚高峰之间列车，晚高峰列车，晚高峰后列车。

设五类列车的数量分别为 $S_i = S_1, S_2, S_3, S_4, S_5$ 。

则：

$$\sum_{i=1}^5 S_i = S_{\text{车}}$$

发车时间间隔  $H=\{h_1, \dots, h_{239}\}$ ,

第 1 类车发车时间间隔为 $\{h_1, \dots, h_{S_1-1}\}$ ;

第 2 类车发车时间间隔为 $\{h_{S_1}, \dots, h_{S_1+S_2-1}\}$ ;

第 3 类车发车时间间隔为 $\{h_{S_1+S_2}, \dots, h_{S_1+S_2+S_3-1}\}$ ,

第 4 类车发车时间间隔为 $\{h_{S_1+S_2+S_3}, \dots, h_{S_1+S_2+S_3+S_4-1}\}$ ,

第 5 类车发车时间间隔为 $\{h_{S_1+S_2+S_3+S_4}, \dots, h_{239}\}$

高峰时间（早高峰 7200 秒至 12600 秒，晚高峰 43200 至 50400 秒）发车间隔不大于 2.5 分钟且不小于 2 分钟，其余时间发车间隔不小于 5 分钟。

所以发车间隔时间约束条件为：

$$\begin{cases} h_i \geq 300, & i \in [1, S_1] \cup [h_{S_1+S_2}, h_{S_1+S_2+S_3-1}] \cup [h_{S_1+S_2+S_3+S_4}, h_{239}] \\ 120 \leq h_i \leq 150, & i \in [h_{S_1}, h_{S_1+S_2-1}] \cup [h_{S_1+S_2+S_3}, h_{S_1+S_2+S_3+S_4-1}] \end{cases}$$

同样，总间隔时间约束应满足：

$$\sum_{i=1}^{239} h_i = T_0$$

$T_0$ 为总间隔时间，且 $T_0=63900s$ 。

## (2) 目标函数

同上一节可再生能量利用下的多列车运行优化模型，论文记第 $i$ 辆列车第 $j$ 次制动产生的被利用的再生能量为 $E_{used(ij)}$ 。则总的被利用的再生能量为：

$$\sum_{i=1}^{239} \sum_{j=1}^{13} E_{used(ij)}。$$

所以优化目标为:  $\max \sum_{i=1}^{239} \sum_{j=1}^{13} E_{used(ij)}$

根据目标的优化和约束，可得列车的节能优化模型：

$$\begin{aligned} & \max \sum_{i=1}^{239} \sum_{j=1}^{13} E_{used(ij)} \\ \text{s.t.} & \begin{cases} \sum_{i=1}^{239} h_i = T_0 \\ \sum_{i=1}^5 S_i = S_{\text{车}} \\ h_i \geq 300, i \in [1, S_1] \cup [h_{S_1+S_2}, h_{S_1+S_2+S_3-1}] \cup [h_{S_1+S_2+S_3+S_4}, h_{239}] \\ 120 \leq h_i \leq 150, i \in [h_{S_1}, h_{S_1+S_2-1}] \cup [h_{S_1+S_2+S_3}, h_{S_1+S_2+S_3+S_4-1}] \end{cases} \end{aligned}$$

## 6.3.2 模型的求解

### (1) 高峰期和非高峰期发车数量分配。

对于考虑高峰期的多列车节能运行优化模型求解，论文首先根据不同时期发车间隔时间的约束条件，确定出不同时期分配的列车数量 $\{S_1, S_2, S_3, S_4, S_5\}$ 。

由高峰时间（早高峰 7200 秒至 12600 秒，晚高峰 43200 至 50400 秒）发车间隔不大于 2.5 分钟且不小于 2 分钟，为了缓解高峰期客流高峰，同时出于安全考虑，高峰期取发车间隔时间 135 秒。其余时间发车间隔不小于 5 分钟。

$$\text{可以求出: } \begin{cases} S_1 \leq 24 \\ S_2 = 40 \\ S_3 \leq 102 \\ S_4 = 54 \\ S_5 \leq 45 \end{cases}$$

$$\text{又: } S_1+S_2+S_3+S_4+S_5 = S_{\text{车}} = 240$$

所以：  $S_1+S_3+S_5=S_{\text{车}} - S_2 - S_4=135$

对于{ $S_1,S_3,S_5$ }的分配，我们按照时间占比进行合理分配，最后的分配方案是：{ $S_1,S_2,S_3,S_4,S_5$ }={21,40,87,54,38}辆。

(2) 高峰期和非高峰期发车间隔时间策略

首先求出各个时期平均发车间隔时间{343,135,351 ,135,355 }秒。

根据列车数量分配方案和各个时期平均发车间隔时间，按照和前一节相同的策略调整发车时间，逐步确定 $H = \{h_1,h_2,\cdots,h_{239}\}$ 。

(3) 模型的结果与分析

按照上面的思路进行分析求解，得到列车在早高峰前，早高峰中，早高峰与晚高峰之间，晚高峰，晚高峰后的发车间隔时间安排如下面的表格。

表 6-2 早高峰前 21 辆车发车间隔

车次	1	2	3	4	5	6	7
间隔		352	348	330	343	347	357
车次	8	9	10	11	12	13	14
间隔	324	351	351	344	345	341	339
车次	15	16	17	18	19	20	21
间隔	342	352	345	352	327	323	358

表 6-3 早高峰 40 辆车发车间隔

车次	22	23	24	25	26	27	28	29
间隔	125	149	147	135	122	130	136	145
车次	30	31	32	33	34	35	36	37
间隔	143	142	145	134	139	127	149	129
车次	38	39	40	41	42	43	44	45
间隔	145	121	121	131	149	136	126	127
车次	46	47	48	49	50	51	52	53
间隔	140	127	134	132	130	137	123	133
车次	22	23	24	25	26	27	28	29
间隔	125	149	147	135	122	130	136	145

表 6-4 早高峰后与晚高峰前 87 辆车发车间隔

车次	62	63	64	65	66	67	68	69	70	71	72
间隔	343	336	369	340	363	334	347	330	332	365	356
车次	73	74	75	76	77	78	79	80	81	82	83
间隔	349	362	362	335	349	345	367	364	367	350	367

车次	84	85	86	87	88	89	90	91	92	93	94
间隔	368	361	362	350	331	350	366	364	356	357	331
车次	95	96	97	98	99	100	101	102	103	104	105
间隔	349	345	334	358	334	361	359	348	339	345	349
车次	106	107	108	109	110	111	112	113	114	115	116
间隔	344	361	353	367	353	330	338	332	369	333	342
车次	117	118	119	120	121	122	123	124	125	126	127
间隔	346	350	337	369	353	364	347	352	330	330	330
车次	128	129	130	131	132	133	134	135	136	137	138
间隔	363	359	362	369	367	359	364	357	366	369	346
车次	139	140	141	142	143	144	145	146	147	148	
间隔	357	332	338	356	364	368	356	369	335	365	

表 6-5 晚高峰 54 辆车发车间隔

车次	149	150	151	152	153	154	155	156	157
间隔	148	132	147	142	134	144	131	139	124
车次	158	159	160	161	162	163	164	165	166
间隔	127	135	120	133	133	120	138	145	140
车次	167	168	169	170	171	172	173	174	175
间隔	143	124	124	130	126	132	122	145	138
车次	176	177	178	179	180	181	182	183	184
间隔	126	121	127	142	123	141	141	121	124
车次	185	186	187	188	189	190	191	192	193
间隔	143	139	121	135	128	132	128	142	149
车次	194	195	196	197	198	199	200	201	202
间隔	148	134	144	121	124	139	133	137	121

表 6-6 晚高峰后 38 辆车发车间隔

车次	203	204	205	206	207	208	209	210
间隔	368	355	343	332	332	377	356	340
车次	211	212	213	214	215	216	217	218
间隔	370	351	334	373	340	361	359	369
车次	219	220	221	222	223	224	225	226
间隔	373	377	338	371	351	354	352	357
车次	227	228	229	230	231	232	233	234
间隔	371	370	345	351	337	330	370	333
车次	235	236	237	238	239	240		
间隔	352	338	375	365	376	354		

由求解的一天之中不同阶段的发车间隔时间，可以验证满足不同阶段的发车间隔时间要求，并且总的间隔时间等于 $T_0=63900s$ 。

## 七、问题三：列车延误后运行优化控制问题

### 7.1 列车站点延误的影响分析

列车晚点是一种常见的发生在轨道交通运行时的现象，主要包括出站晚点和站间运行中的晚点两种情况。列车经常由于车站客流、司机操作等原因造成出站延误，这里研究列车的出站延误问题。

设相邻列车的最小安全距离为 $L_{safe}$ ，对于出发间隔时间为 $h_i$ 的前后两辆车，在整条线路上存在一个最小距离处 $L_{min}$ ，一般会有 $L_{min} \geq L_{safe}$ 。当出发间隔时间增大时存在的最小距离也会相应增大，更加安全。当出发间隔时间减小时，最小距离 $L_{min}$ 也会减小。

列车 $i$ 在站点 $j$ 出发延迟时间 $DT_j^i$ ，在特点上与减小发车间隔相同。即：使前后列车在线路上的最小距离 $L_{min}$ 减小，延迟时间的大小也对应着最小距离 $L_{min}$ 减少的程度。当延迟时间较小时，最小距离若仍能保持关系 $L_{min} \geq L_{safe}$ 则后面列车不是必须进行延迟调整的；当延迟时间较大，最小距离恶化严重，使 $L_{min} < L_{safe}$ ，则后面列车必须进行延迟调整。所以， $DT_j^i$ 的大小与后面列车需要进行延迟调整的程度，迫切性具有一定的相关性。假设后面列车延迟调整的时间 $DT_{j-1}^{i+1}=f(DT_j^i)$ ，则由 $DT_j^i$ 的概率分布可以由关系 $DT_{j-1}^{i+1}=f(DT_j^i)$ 求得。

当 $DT_j^i < 10s$ 时，为普通延误，发生概率为20%；当 $DT_j^i > 10s$ 时，为严重失误，发生概率为10%。 $DT_j^i = 0$ 的概率为70%。

当为严重失误时， $DT_{j-1}^{i+1}$ 需要调整的程度和概率均增加；

当为普通失误是， $DT_{j-1}^{i+1}$ 需要调整的程度和概率均较小。

针对不同时段，比如高峰期与正常时间段相比，发车间隔较小， $L_{min}$ 已经较小，这时对于相同的 $DT_j^i$ ，在高峰期进行延迟调整的可能性和调整程度均较正常时段大。所以在高峰时段后续列车调整时应该较正常时调整更多。

调整发车间隔和在站点停车的时间，再生能量利用的情况也不同。在站点停车的时间也会影响列车在站间运行的时间，进而影响能耗。当 $DT_j^i$ 较大时，需要调整的时间也增加，能耗也增加。可以根据 $DT_j^i$ 的分布特点，推到需要调整的时间和能耗变化。我们可以据此研究最佳调整方案。

## 7.2 基于“延点-赶点结合”调整策略的列车延误后运行优化控制模型

当发生列车出站延误时，应采取一定的调整策略尽快使所有后续列车恢复正点运行，同时降低调整期间的耗能。

查阅相关文献知，目前已经提出的调整策略包括：赶点运行调整，延点运行调整和延点-赶点结合调整等策略。其中延点-赶点结合策略综合了延点和赶点调整策略。赶点策略是指发生站点延误的列车在下一站间将时间赶回来，甚至跳过下一站点（不再下一站点停车），延点策略是指后续列车在得到发生站点延误信息后，延迟机车调整列策划追踪间隔等方面。

这里研究基于“延点-赶点结合”调整策略的列车延误后运行优化控制问题。

### 7.2.1 模型的建立

当列车  $i$  在车站  $A_j$  延误  $DT_j^i$  发车时，采用“延点-赶点结合”策略对线路上的列车进行调整，即先采用延点策略再采用赶点策略控制列车运行。

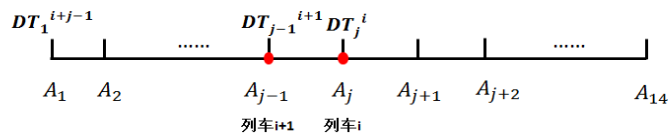


图 7-1 延点策略示意图

延点策略：如图\*\*\*所示，在列车  $i$  在车站  $A_j$  延误  $DT_j^i$  发生后，后续列车  $i+1, i+2, \dots, i+j-1$  分别在从车站  $A_{j-1}, A_{j-2}, \dots, A_1$  出站时调整延迟

$DT_{j-1}^{i+1}, DT_{j-2}^{i+2}, \dots, DT_1^{i+j-1}$ , 并且为了防止后续列车的延误恶化晚点情况, 有如下基本约束条件:

$$DT_j^i \geq DT_{j-1}^{i+1} \geq DT_{j-2}^{i+2} \dots \geq DT_1^{i+j-1}。$$

赶点策略: 在站点延迟的列车在接下来的站间运行时减少站间时间。为了使列车尽快恢复正点运行, 考虑利用两站路程进行调整, 即在接下来的两站分别减少站间时间。以列车  $i$  为例, 其对应的赶点策略为在  $A_j - A_{j+1}$  站间减少运行时间, 记为  $DT_j^i(1)$ , 在  $A_{j+1} - A_{j+2}$  站间减少运行时间, 记为  $DT_{j+1}^i(2)$ , 且满足:

$$DT_j^i(1) + DT_{j+1}^i(2) = DT_j^i$$

同理:

$$DT_{j-1}^{i+1}(1) + DT_{j+1}^{i+1}(2) = DT_{j-1}^{i+1}$$

$$DT_{j-2}^{i+2}(1) + DT_{j+2}^{i+2}(2) = DT_{j-2}^{i+2}$$

.....

$$DT_1^{i+j-1}(1) + DT_{j+1}^{i+j-1}(2) = DT_1^{i+j-1}$$

列车  $i$  在  $A_j - A_{j+1}$  站间减少运行时间  $DT_j^i(1)$ , 耗能增加记为  $\Delta E_i(1)$ , 在  $A_{j+1} - A_{j+2}$  站间减少运行时间  $DT_{j+1}^i(2)$ , 耗能记为  $\Delta E_i(2)$ 。同理, 记列车  $k$  第一次调整时间, 耗能增加记为  $\Delta E_k(1)$ , 第二次调整时间耗能增加记为  $\Delta E_k(2)$ 。

另外, 在采用赶点策略时, 需要考虑对前面列车运行的影响, 速度不能超过限制速度  $V_{limit}$  防止追尾。  $V_{limit}$  计算公式如下:

$$V_{limit} = \min(V_{line}, \sqrt{2LB_e})$$

其中:  $V_{line}$  是列车当前位置线路限速,  $L$  是当前时刻前后车之间距离,  $B_e$  是列车制动的最大减速度。

采用“延点-赶点结合”调整策略的优化目标函数为:

$$\min \sum_{k=i}^{i+j-1} [\Delta E_k(1) + \Delta E_k(2)]$$

## 7.2.2 模型的求解

论文首先假设某一特定的列车出站延误情景, 按照“延点-赶点结合”调整策略对列车运行控制。

对于非高峰时段，列车的发车时间间隔较大，可以认为前一列车发生延误对后续车辆不会产生影响。因此本节以问题二考虑早高峰时段的多列车运行情况为例，假设 A 车在 A2 站到站时延误 10s，且后续列车 B 车由于受到 A 车影响在到达 A1 站时也有一定的延误，为了“使所有后续列车尽快恢复正点运行”，那么 A、B 列车就应该尽快的将延误的时间通过减少接下来几站路的耗时而弥补回来。因此约定，A2 站延迟出发的列车 A，最晚将在 A2-A4 路段上把延误的时间全部弥补回来；而在 A1 站延迟出发的列车 B，将在 A1-A2 站把延误的时间弥补回来。

假如 A 车在 A2-A3 段弥补的时间为  $t_1$ ，那么 A 车在 A3-A4 需要段弥补的时间为  $10 - t_1$ ，则 B 车在 A1-A2 段需要弥补的时间也为  $10 - t_1$ 。本文按照以下步骤找出该背景下的列车最优控制方案：

Step1: 计算列车 A 在 A2-A3 路段上减少  $t_1 (t_1 = 1, 2, \dots, 10)$  时间而增加的能  $\Delta E(A2 - A3)$ 。

Step2: 计算列车 A 在 A3-A4 路段上减少  $10 - t_1$  时间而增加的能耗  $\Delta E(A3 - A4)$ 。

Step3: 计算列车 B 在 A1-A2 路段上减少  $10 - t_1$  时间而增加的能耗  $\Delta E(A1 - A2)$ 。

Step4: 对于  $t_1$  每一个可能的取值，计算  $E = \Delta E(A1 - A2) + \Delta E(A2 - A3) + \Delta E(A3 - A4)$ ，找到使 E 最小的  $t_1$  值，作为该问题的最佳解决方案。

经计算，结果如下：

表 7-1 模型探索求解变化

$t_1$	$t_2$	$\Delta E(A1 - A2)$	$\Delta E(A2 - A3)$	$\Delta E(A3 - A4)$	E
-1	-9	0.0325	0.9298	0.0028	0.9651
-2	-8	-0.2798	0.9298	-0.0878	0.5622
-3	-7	-0.2892	0.6907	0.0247	0.4262
-4	-6	0.0206	0.6907	0.3415	1.0528
-5	-5	0.0752	0.6907	0.3143	1.0802
-6	-4	0.0752	0.4314	0.3747	0.8813
-7	-3	0.0898	0.5142	0.3747	0.9787



-8	-2	0.2098	0.2887	0.7343	1.2328
-9	-1	0.5716	0.1499	0.3205	1.042
-10	0	0.708	0	0.5149	1.2229

$t_1, t_2$  单位为 s, E 单位为  $10^7 J$ 。

由上表可知，当  $t_1 = -3s$  时，增加的能耗最小，即当 A 列车在 A2-A3 段间弥补 3 秒，在 A3-A4 段间弥补 7 秒，B 车在 A1-A2 段间弥补 7 秒时，两车所需要增加的耗能最小。

A 车在 A2-A4 站的运行规律如图 7-2 所示。

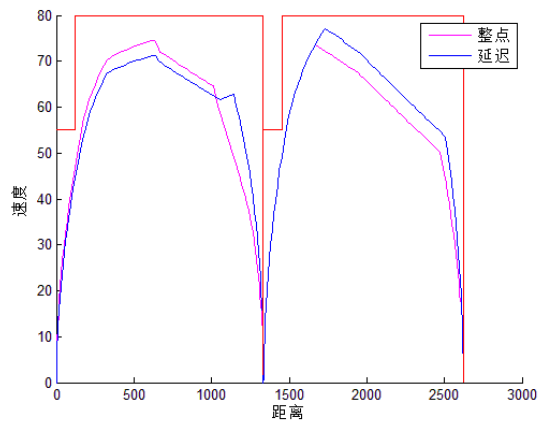


图 7-2 A 车在 A2-A4 站间的距离—速度曲线

### 7.3 延误时间为随机变量时对控制方案的影响分析

当  $DT_j^i$  为随机变量时，因为允许列车在各站到、发时间与原时间相比提前不超过 10 秒，但可以达到 10 秒，因此普通延误情形  $DT_j^i < 10$  秒下，该列车在下一段路可以直接追回。这种列车直接在下一段路赶上正点的概率为 20%。当严重延误发生时，先将后续车辆延误时间都在一段路追回 10 秒，再继续调整第二问的控制方案。

## 八、模型评价

### 8.1 模型的优点

(1) 模型建立过程逻辑性强，层层推进，清晰明确

(2) 问题理解深入, 对于难以求解的优化问题, 结合生活实际合理假设, 巧妙处理, 使问题处理比较方便使用。

(2) 针对问题一和问题二建立的模型及算法设计, 求解策略等解决问题效果好, 结果明确。

## 8.2 模型的缺点

(1) 在模型求解方面, 因为问题的复杂性, 在不采用智能算法处理问题的情况下求解效率相对较低。可以采用遗传算法、蚁群算法等求解模型;

(2) 对于列车延迟具有随机性, 相应的运行优化控制问题研究不够。

## 九、参考文献

- [1] 李坤妃. 多列车协同控制节能优化方法的研究[D].北京:北京交通大学, 2014.
- [2] 王青元,冯晓云,朱金陵,梁志成. 考虑再生制动能量利用的高速列车节能最优控制仿真研究[J]. 中国铁道科学,2015,01:96-103.
- [3] 冯佳. 考虑节能目标的城市轨道交通列车运行行为优化研究[D].北京交通大学,2014
- [4] 顾青. 城市轨道交通列车节能优化驾驶研究[D].北京:北京交通大学, 2014.
- [5] 李娜. 城市轨道交通车辆运行节能方法优化研究[D].上海:上海工程技术大学, 2010.
- [6] 石红国. 列车运行过程仿真及优化研究[D].西南交通大学, 2006.
- [7] 李欢. 地铁列车运行仿真及策略优化[D].西南交通大学, 2012.[6] 赵乐. 基于再生制动的地铁列车时刻表优化模型与算法研究[D].北京交通大学, 2014.
- [8] 吴洋. 晚点情况下地铁列车实时运行调整及速度控制模式研究[D].西南交通大学,2004.
- [9] 吴洋,罗霞. 一种晚点地铁列车实时调整策略及其动态速控模式[J]. 中国铁道科学,2005,06:115-120.

## 十、附录

### 10.1 数据输出

#### 1.问题 1-1： A6-A7 站不考虑巡航时列车运行参数

车站名称	A6	A7
发车时刻	0:00:00	
站间实际长度(m)	1354	
停站时间(s)	XXX	
站间运行时间(s)	110	
满载率(%)	70%	

时刻(hh:mm:ss)	实际速度(cm/s)	实际速度(km/h)	计算加速度 (m/s <sup>2</sup> )	计算距离(m)	计算公里标(m)	当前坡度(%)	计算牵引力(N)	计算牵引功率 (Kw)
0:00:00	0	0	0	0	13594	0	0	0
0:00:01	60.69780034	2.185120812	0.606978003	0.606978003	13593.39302	0	117936.6584	71.58495744
0:00:02	121.253949	4.365142166	0.605561487	1.819517494	13592.18048	0	117661.7115	142.6694718
0:00:03	181.651925	6.5394693	0.603979759	3.636036744	13590.36396	0	117354.6971	213.1770664
0:00:04	241.8753752	8.707513508	0.602234502	6.054790496	13587.94521	0	117015.9415	283.0327475
0:00:05	301.9081324	10.86869277	0.600327571	9.07387182	13584.92613	0	116645.8048	352.1631709
0:00:06	361.7342317	13.02243234	0.598260993	12.69121414	13581.30879	0	116244.6805	420.4968019

0:00:07	421.3379276	15.16816539	0.596036959	16.90459341	13577.09541	0	115812.9939	487.9640685
0:00:08	480.7037097	17.30533355	0.593657822	21.71163051	13572.28837	0	115351.2017	554.4975057
0:00:09	539.8163186	19.43338747	0.591126089	27.1097937	13566.89021	0	114859.7906	620.0318932
0:00:10	598.6607605	21.55178738	0.588444419	33.0964013	13560.9036	0	114339.2766	684.5043828
0:00:11	657.2223219	23.66000359	0.585615614	39.66862452	13554.33138	0	113790.2035	747.8546172
0:00:12	715.4865831	25.75751699	0.582642612	46.82349035	13547.17651	0	113213.1418	810.0248399
0:00:13	773.4394317	27.84381954	0.579528486	54.55788467	13539.44212	0	112608.6878	870.9599948
0:00:14	831.0670749	29.9184147	0.576276432	62.86855542	13531.13144	0	111977.4617	930.6078153
0:00:15	888.3560511	31.98081784	0.572889762	71.75211593	13522.24788	0	111320.1068	988.9189046
0:00:16	945.2932415	34.03055669	0.569371903	81.20504834	13512.79495	0	110637.2878	1045.846805
0:00:17	1001.86588	36.06717167	0.565726383	91.22370714	13502.77629	0	109929.6898	1101.348054
0:00:18	1058.061563	38.09021625	0.561956828	101.8043228	13492.19568	0	109198.0164	1155.382239
0:00:19	1113.868258	40.09925728	0.558066951	112.9430053	13481.05699	0	108442.9886	1207.912028
0:00:20	1169.274313	42.09387525	0.554060549	124.6357485	13469.36425	0	107665.3432	1258.903202
0:00:21	1224.268462	44.07366463	0.549941493	136.8784331	13457.12157	0	106865.8315	1308.324671
0:00:22	1278.839834	46.03823401	0.545713719	149.6668314	13444.33317	0	106045.2176	1356.148484
0:00:23	1332.977956	47.98720642	0.541381224	162.996611	13431.00339	0	105204.2772	1402.349823
0:00:24	1386.672761	49.92021941	0.536948054	176.8633386	13417.13666	0	104343.7958	1446.906995
0:00:25	1439.914592	51.8369253	0.532418302	191.2624845	13402.73752	0	103464.5678	1489.801409
0:00:26	1492.694201	53.73699125	0.527796096	206.1894265	13387.81057	0	102567.3943	1531.017547
0:00:27	1545.00276	55.62009938	0.523085592	221.6394541	13372.36055	0	101653.0822	1570.542927
0:00:28	1588.065013	57.17034047	0.430622525	237.5201043	13356.4799	0	83688.90209	1329.034174
0:00:29	1627.88143	58.60373148	0.398164172	253.7989186	13340.20108	0	77383.19159	1259.706606

0:00:30	1664.87997	59.9356789	0.369985394	270.4477183	13323.55228	0	71908.93681	1197.197485
0:00:31	1699.40795	61.1786862	0.345279804	287.4417978	13306.5582	0	67109.46507	1140.463584
0:00:32	1731.750806	62.34302902	0.323428563	304.7593058	13289.24069	0	62864.54336	1088.657236
0:00:33	1760.381744	63.3737428	0.286309382	322.3631233	13271.63688	1.8	55656.53224	979.7674331
0:00:34	1787.363286	64.3450783	0.269815416	340.2367561	13253.76324	1.8	52452.40515	937.5150322
0:00:35	1812.85102	65.26263673	0.254877342	358.3652663	13235.63473	1.8	49550.55389	898.2777216
0:00:36	1836.978457	66.13122445	0.241274368	376.7350509	13217.26495	1.8	46908.08183	861.6913579
0:00:37	1859.860991	66.95499567	0.228825339	395.3336608	13198.66634	1.8	44489.79328	827.4483102
0:00:38	1844.086587	66.38711714	-0.157744035	413.7745267	13180.22547	1.8	0	0
0:00:39	1828.480893	65.82531214	-0.156056946	432.0593356	13161.94066	1.8	0	0
0:00:40	1813.040979	65.26947523	-0.15439914	450.1897454	13143.81025	1.8	0	0
0:00:41	1797.763984	64.71950343	-0.152769946	468.1673852	13125.83261	1.8	0	0
0:00:42	1782.647113	64.17529606	-0.151168713	485.9938564	13108.00614	1.8	0	0
0:00:43	1767.687632	63.63675475	-0.149594807	503.6707327	13090.32927	1.8	0	0
0:00:44	1752.882871	63.10378334	-0.148047614	521.1995614	13072.80044	1.8	0	0
0:00:45	1738.230217	62.57628781	-0.146526536	538.5818636	13055.41814	1.8	0	0
0:00:46	1723.727118	62.05417624	-0.145030993	555.8191347	13038.18087	1.8	0	0
0:00:47	1709.371076	61.53735874	-0.143560418	572.9128455	13021.08715	1.8	0	0
0:00:48	1695.15965	61.02574739	-0.142114263	589.864442	13004.13556	1.8	0	0
0:00:49	1681.09045	60.51925622	-0.140691993	606.6753465	12987.32465	1.8	0	0
0:00:50	1667.161142	60.0178011	-0.139293089	623.3469579	12970.65304	1.8	0	0
0:00:51	1653.369437	59.52129974	-0.137917044	639.8806523	12954.11935	1.8	0	0
0:00:52	1639.7131	59.02967161	-0.136563368	656.2777833	12937.72222	1.8	0	0

0:00:53	1626.189942	58.54283792	-0.135231581	672.5396827	12921.46032	1.8	0	0
0:00:54	1612.79782	58.06072154	-0.133921217	688.6676609	12905.33234	1.8	0	0
0:00:55	1604.728638	57.77023097	-0.080691823	704.7149473	12889.28505	-3.5	0	0
0:00:56	1596.736749	57.48252295	-0.079918896	720.6823148	12873.31769	-3.5	0	0
0:00:57	1588.821117	57.1975602	-0.079156319	736.5705259	12857.42947	-3.5	0	0
0:00:58	1580.980726	56.91530613	-0.078403908	752.3803332	12841.61967	-3.5	0	0
0:00:59	1573.214577	56.63572479	-0.077661484	768.112479	12825.88752	-3.5	0	0
0:01:00	1565.52169	56.35878085	-0.076928872	783.7676959	12810.2323	-3.5	0	0
0:01:01	1557.9011	56.08443961	-0.0762059	799.3467069	12794.65329	-3.5	0	0
0:01:02	1550.35186	55.81266696	-0.075492401	814.8502255	12779.14977	-3.5	0	0
0:01:03	1542.873039	55.5434294	-0.074788211	830.2789559	12763.72104	-3.5	0	0
0:01:04	1535.463722	55.276694	-0.074093169	845.6335931	12748.36641	-3.5	0	0
0:01:05	1528.12301	55.01242837	-0.073407117	860.9148232	12733.08518	-3.5	0	0
0:01:06	1520.85002	54.75060072	-0.072729903	876.1233234	12717.87668	-3.5	0	0
0:01:07	1513.643882	54.49117977	-0.072061376	891.2597622	12702.74024	-3.5	0	0
0:01:08	1506.503744	54.23413477	-0.071401389	906.3247997	12687.6752	-3.5	0	0
0:01:09	1499.428764	53.9794355	-0.070749797	921.3190873	12672.68091	-3.5	0	0
0:01:10	1492.418118	53.72705224	-0.07010646	936.2432685	12657.75673	-3.5	0	0
0:01:11	1485.470994	53.47695578	-0.069471238	951.0979784	12642.90202	-3.5	0	0
0:01:12	1478.586594	53.22911739	-0.068843998	965.8838444	12628.11616	-3.5	0	0
0:01:13	1471.764134	52.98350881	-0.068224607	980.6014857	12613.39851	-3.5	0	0
0:01:14	1465.00284	52.74010224	-0.067612934	995.2515141	12598.74849	-3.5	0	0
0:01:15	1458.301955	52.49887037	-0.067008853	1009.834534	12584.16547	-3.5	0	0

0:01:16	1451.660731	52.25978631	-0.06641224	1024.351141	12569.64886	-3.5	0	0
0:01:17	1445.078434	52.02282361	-0.065822972	1038.801925	12555.19807	-3.5	0	0
0:01:18	1438.554341	51.78795626	-0.06524093	1053.187469	12540.81253	-3.5	0	0
0:01:19	1432.087741	51.55515867	-0.064665997	1067.508346	12526.49165	-3.5	0	0
0:01:20	1425.677935	51.32440566	-0.064098059	1081.765125	12512.23487	-3.5	0	0
0:01:21	1419.324235	51.09567245	-0.063537003	1095.958368	12498.04163	-3.5	0	0
0:01:22	1413.025963	50.86893465	-0.06298272	1110.088627	12483.91137	-3.5	0	0
0:01:23	1406.782453	50.64416829	-0.0624351	1124.156452	12469.84355	-3.5	0	0
0:01:24	1400.593049	50.42134975	-0.061894039	1138.162382	12455.83762	-3.5	0	0
0:01:25	1394.457105	50.20045579	-0.061359433	1152.106953	12441.89305	-3.5	0	0
0:01:26	1388.373987	49.98146354	-0.060831181	1165.990693	12428.00931	-3.5	0	0
0:01:27	1382.343069	49.76435049	-0.060309182	1179.814124	12414.18588	-3.5	0	0
0:01:28	1376.363735	49.54909447	-0.059793339	1193.577761	12400.42224	-3.5	0	0
0:01:29	1370.43538	49.33567367	-0.059283556	1207.282115	12386.71788	-3.5	0	0
0:01:30	1364.557406	49.12406661	-0.05877974	1220.927689	12373.07231	-3.5	0	0
0:01:31	1358.729226	48.91425213	-0.058281798	1234.514982	12359.48502	-3.5	0	0
0:01:32	1284.600587	46.24562115	-0.741286384	1247.360987	12346.63901	-3.5	0	0
0:01:33	1211.084322	43.59903559	-0.735162655	1259.471831	12334.52817	-3.5	0	0
0:01:34	1138.15046	40.97341655	-0.729338624	1270.853335	12323.14666	-3.5	0	0
0:01:35	1065.769873	38.36771542	-0.723805868	1281.511034	12312.48897	-3.5	0	0
0:01:36	993.9142255	35.78091212	-0.718556473	1291.450176	12302.54982	-3.5	0	0
0:01:37	922.5559243	33.21201328	-0.713583012	1300.675735	12293.32426	-3.5	0	0
0:01:38	851.6680728	30.66005062	-0.708878515	1309.192416	12284.80758	-3.5	0	0

0:01:39	777.7944274	28.00059939	-0.738736454	1316.97036	12277.02964	0	0	0
0:01:40	704.3591547	25.35692957	-0.734352727	1324.013952	12269.98605	0	0	0
0:01:41	631.3348265	22.72805375	-0.730243282	1330.3273	12263.6727	0	0	0
0:01:42	558.6945975	20.11300551	-0.72640229	1335.914246	12258.08575	0	0	0
0:01:43	486.4121634	17.51083788	-0.722824341	1340.778368	12253.22163	0	0	0
0:01:44	414.4617205	14.92062194	-0.719504429	1344.922985	12249.07701	0	0	0
0:01:45	342.817927	12.34144537	-0.716437935	1348.351164	12245.64884	0	0	0
0:01:46	271.4558654	9.772411155	-0.713620616	1351.065723	12242.93428	0	0	0
0:01:47	200.3510067	7.21263624	-0.711048587	1353.069233	12240.93077	0	0	0
0:01:48	129.4791749	4.661250295	-0.708718318	1354.364025	12239.63598	0	0	0
0:01:49	58.8165133	2.117394479	-0.706626616	1354.95219	12239.04781	0	0	0
0:01:50	0	0	-0.704770618	1354.835584	12239.16442	0	0	0



## 2.问题 1-1： A6-A7 站考虑巡航时列车运行参数

车站名称	A6	A7
发车时刻	0:00:00	
站间实际长度(m)	1354	
停站时间(s)	XXX	
站间运行时间(s)	110	
满载率(%)	70%	

时刻(hh:mm:ss)	实际速度(cm/s)	实际速度(km/h)	计算加速度 (m/s <sup>2</sup> )	计算距离(m)	计算公里标(m)	当前坡度(‰)	计算牵引力(N)	计算牵引功率 (Kw)
0:00:00	0	0	0	0	13594	0	0	0
0:00:01	100	3.6	1	1.5	13592.5	0	194295	194.295
0:00:02	200	7.2	1	4	13590	0	194295	388.59
0:00:03	300	10.8	1	7.5	13586.5	0	194295	582.885
0:00:04	391.1770126	14.08237245	0.911770126	11.86765519	13582.13234	0	182700	714.680402
0:00:05	482.0093131	17.35233527	0.908323005	17.14190982	13576.85809	0	182700	880.631015
0:00:06	572.4602624	20.60856945	0.904509494	23.31876719	13570.68123	0	182700	1045.884899
0:00:07	662.4938295	23.84977786	0.900335671	30.39387332	13563.60613	0	182700	1210.376226
0:00:08	752.0746467	27.07468728	0.895808172	38.36252388	13555.63748	0	182700	1374.04038
0:00:09	841.1680637	30.28205029	0.89093417	47.2196716	13546.78033	0	182700	1536.814052
0:00:10	929.7401984	33.47064714	0.885721347	56.95993426	13537.04007	0	182700	1698.635342
0:00:11	1017.757986	36.63928748	0.880177872	67.57760305	13526.4224	0	182700	1859.44384

0:00:12	1105.189223	39.78681203	0.874312374	79.06665147	13514.93335	0	182700	2019.180711
0:00:13	1192.002614	42.91209411	0.868133912	91.42074456	13502.57926	0	182700	2177.788776
0:00:14	1278.167809	46.01404112	0.861651946	104.6332486	13489.36675	0	182700	2335.212587
0:00:15	1363.65544	49.09159583	0.85487631	118.6972412	13475.30276	0	182700	2491.398489
0:00:16	1448.437157	52.14373767	0.847817176	133.6055213	13460.39448	0	182700	2646.294687
0:00:17	1532.48566	55.16948376	0.840485027	149.3506205	13444.64938	0	182700	2799.851301
0:00:18	1603.926745	57.74136283	0.714410851	165.7470933	13428.25291	0	159679.9722	2561.149781
0:00:19	1667.643964	60.0351827	0.637172187	182.7421191	13411.25788	0	145976.6382	2434.370596
0:00:20	1667.643964	60.0351827	0	199.4185587	13394.58144	0	23378.46665	389.869588
0:00:21	1667.643964	60.0351827	0	216.0949983	13377.905	0	23378.46665	389.869588
0:00:22	1667.643964	60.0351827	0	232.771438	13361.22856	0	23378.46665	389.869588
0:00:23	1667.643964	60.0351827	0	249.4478776	13344.55212	0	23378.46665	389.869588
0:00:24	1667.643964	60.0351827	0	266.1243173	13327.87568	0	23378.46665	389.869588
0:00:25	1667.643964	60.0351827	0	282.8007569	13311.19924	0	23378.46665	389.869588
0:00:26	1667.643964	60.0351827	0	299.4771965	13294.5228	0	23378.46665	389.869588
0:00:27	1667.643964	60.0351827	0	316.1536362	13277.84636	0	23378.46665	389.869588
0:00:28	1667.643964	60.0351827	0	332.8300758	13261.16992	1.8	26805.83045	447.0258135
0:00:29	1667.643964	60.0351827	0	349.5065155	13244.49348	1.8	26805.83045	447.0258135
0:00:30	1667.643964	60.0351827	0	366.1829551	13227.81704	1.8	26805.83045	447.0258135
0:00:31	1667.643964	60.0351827	0	382.8593947	13211.14061	1.8	26805.83045	447.0258135
0:00:32	1667.643964	60.0351827	0	399.5358344	13194.46417	1.8	26805.83045	447.0258135
0:00:33	1667.643964	60.0351827	0	416.212274	13177.78773	1.8	26805.83045	447.0258135
0:00:34	1667.643964	60.0351827	0	432.8887137	13161.11129	1.8	26805.83045	447.0258135

0:00:35	1653.847505	59.53851017	-0.137964592	449.3582064	13144.64179	1.8	0	0
0:00:36	1640.18649	59.04671365	-0.136610145	465.6917662	13128.30823	1.8	0	0
0:00:37	1626.65873	58.55971427	-0.135277604	481.8907147	13112.10929	1.8	0	0
0:00:38	1613.26208	58.07743487	-0.133966501	497.9563523	13096.04365	1.8	0	0
0:00:39	1599.994441	57.59979989	-0.132676384	513.8899585	13080.11004	1.8	0	0
0:00:40	1586.85376	57.12673537	-0.131406811	529.6927927	13064.30721	1.8	0	0
0:00:41	1573.838025	56.65816891	-0.13015735	545.3660943	13048.63391	1.8	0	0
0:00:42	1560.945267	56.19402961	-0.128927582	560.9110832	13033.08892	1.8	0	0
0:00:43	1548.173557	55.73424805	-0.1277171	576.3289602	13017.67104	1.8	0	0
0:00:44	1535.521006	55.27875623	-0.126525506	591.6209075	13002.37909	1.8	0	0
0:00:45	1522.985765	54.82748755	-0.125352412	606.7880889	12987.21191	1.8	0	0
0:00:46	1510.566021	54.38037676	-0.12419744	621.8316504	12972.16835	1.8	0	0
0:00:47	1498.259999	53.93735996	-0.123060224	636.7527203	12957.24728	1.8	0	0
0:00:48	1486.065959	53.49837451	-0.121940403	651.5524097	12942.44759	1.8	0	0
0:00:49	1473.982196	53.06335904	-0.12083763	666.2318128	12927.76819	1.8	0	0
0:00:50	1462.007039	52.63225342	-0.119751562	680.7920074	12913.20799	1.8	0	0
0:00:51	1450.138853	52.2049987	-0.118681867	695.234055	12898.76594	1.8	0	0
0:00:52	1443.57003	51.9685211	-0.065688223	709.6369112	12884.36309	-3.5	0	0
0:00:53	1437.059248	51.73413291	-0.065107829	723.9749498	12870.02505	-3.5	0	0
0:00:54	1430.605796	51.50180864	-0.064534518	738.2487405	12855.75126	-3.5	0	0
0:00:55	1424.208978	51.27152321	-0.063968176	752.4588462	12841.54115	-3.5	0	0
0:00:56	1417.868109	51.04325193	-0.06340869	766.6058229	12827.39418	-3.5	0	0
0:00:57	1411.582514	50.8169705	-0.062855951	780.6902201	12813.30978	-3.5	0	0

0:00:58	1405.351529	50.59265503	-0.062309852	794.7125804	12799.28742	-3.5	0	0
0:00:59	1399.1745	50.370282	-0.061770288	808.6734403	12785.32656	-3.5	0	0
0:01:00	1393.050784	50.14982824	-0.061237155	822.5733296	12771.42667	-3.5	0	0
0:01:01	1386.979749	49.93127098	-0.060710352	836.4127719	12757.58723	-3.5	0	0
0:01:02	1380.960771	49.71458777	-0.06018978	850.1922847	12743.80772	-3.5	0	0
0:01:03	1374.993237	49.49975654	-0.059675342	863.9123794	12730.08762	-3.5	0	0
0:01:04	1369.076543	49.28675555	-0.059166942	877.5735614	12716.42644	-3.5	0	0
0:01:05	1363.210094	49.07556339	-0.058664487	891.1763301	12702.82367	-3.5	0	0
0:01:06	1357.393306	48.866159	-0.058167886	904.7211792	12689.27882	-3.5	0	0
0:01:07	1351.625601	48.65852163	-0.057677048	918.2085967	12675.7914	-3.5	0	0
0:01:08	1345.906412	48.45263084	-0.057191886	931.6390648	12662.36094	-3.5	0	0
0:01:09	1340.235181	48.24846652	-0.056712312	945.0130605	12648.98694	-3.5	0	0
0:01:10	1334.611357	48.04600884	-0.056238242	958.3310549	12635.66895	-3.5	0	0
0:01:11	1329.034397	47.84523831	-0.055769593	971.5935141	12622.40649	-3.5	0	0
0:01:12	1323.503769	47.64613569	-0.055306283	984.8008987	12609.1991	-3.5	0	0
0:01:13	1318.018946	47.44868205	-0.054848232	997.953664	12596.04634	-3.5	0	0
0:01:14	1312.57941	47.25285875	-0.054395361	1011.05226	12582.94774	-3.5	0	0
0:01:15	1307.184651	47.05864742	-0.053947593	1024.097133	12569.90287	-3.5	0	0
0:01:16	1301.834165	46.86602995	-0.053504852	1037.088722	12556.91128	-3.5	0	0
0:01:17	1296.527459	46.67498852	-0.053067065	1050.027463	12543.97254	-3.5	0	0
0:01:18	1291.264043	46.48550555	-0.052634157	1062.913787	12531.08621	-3.5	0	0
0:01:19	1286.043437	46.29756374	-0.052206058	1075.748118	12518.25188	-3.5	0	0
0:01:20	1280.865168	46.11114604	-0.051782697	1088.530878	12505.46912	-3.5	0	0

0:01:21	1275.728767	45.92623562	-0.051364005	1101.262484	12492.73752	-3.5	0	0
0:01:22	1270.633776	45.74281593	-0.050949914	1113.943347	12480.05665	-3.5	0	0
0:01:23	1265.57974	45.56087064	-0.050540358	1126.573874	12467.42613	-3.5	0	0
0:01:24	1260.566213	45.38038366	-0.050135271	1139.154469	12454.84553	-3.5	0	0
0:01:25	1255.592754	45.20133914	-0.04973459	1151.685529	12442.31447	-3.5	0	0
0:01:26	1250.658929	45.02372144	-0.04933825	1164.167449	12429.83255	-3.5	0	0
0:01:27	1245.76431	44.84751515	-0.048946191	1176.600619	12417.39938	-3.5	0	0
0:01:28	1240.908475	44.67270509	-0.048558352	1188.985425	12405.01458	-3.5	0	0
0:01:29	1236.091007	44.49927627	-0.048174672	1201.322247	12392.67775	-3.5	0	0
0:01:30	1231.311498	44.32721393	-0.047795093	1213.611465	12380.38854	-3.5	0	0
0:01:31	1226.569542	44.15650352	-0.047419558	1225.85345	12368.14655	-3.5	0	0
0:01:32	1221.864741	43.98713069	-0.04704801	1238.048574	12355.95143	-3.5	0	0
0:01:33	1217.196702	43.81908127	-0.046680393	1250.197201	12343.8028	-3.5	0	0
0:01:34	1212.565037	43.65234132	-0.046316652	1262.299693	12331.70031	-3.5	0	0
0:01:35	1207.969363	43.48689708	-0.045956735	1274.356408	12319.64359	-3.5	0	0
0:01:36	1203.409305	43.32273496	-0.045600587	1286.367701	12307.6323	-3.5	0	0
0:01:37	1113.447396	40.08410624	-0.899619089	1297.052365	12296.94763	-3.5	0	0
0:01:38	1024.161252	36.86980507	-0.892861436	1306.847547	12287.15245	-3.5	0	0
0:01:39	932.079067	33.55484641	-0.920821851	1315.707927	12278.29207	0	0	0
0:01:40	840.6123643	30.26204511	-0.914667027	1323.656717	12270.34328	0	0	0
0:01:41	749.7184997	26.98986599	-0.908938646	1330.699432	12263.30057	0	0	0
0:01:42	659.3558443	23.73681039	-0.903626554	1336.841178	12257.15882	0	0	0
0:01:43	569.4837036	20.50141333	-0.898721407	1342.086654	12251.91335	0	0	0

0:01:44	480.0622399	17.28224063	-0.894214638	1346.440169	12247.55983	0	0	0
0:01:45	391.0523974	14.07788631	-0.890098424	1349.905644	12244.09436	0	0	0
0:01:46	302.4158315	10.88696993	-0.886365659	1352.486619	12241.51338	0	0	0
0:01:47	214.1148391	7.708134208	-0.883009924	1354.186263	12239.81374	0	0	0
0:01:48	126.1122928	4.540042539	-0.880025463	1355.007373	12238.99263	0	0	0
0:01:49	38.3715759	1.381376733	-0.877407169	1354.952385	12239.04761	0	0	0
0:01:50	0	0	-0.875150554	1354.023375	12239.97662	0	0	0

3.问题 1-2： A6-A8 站列车运行参数

车站名称	A6	A7
发车时刻	0:00:00	
站间实际长度(m)	1354	
停站时间(s)	XXX	
站间运行时间(s)	110	
满载率(%)	70%	

时刻 (hh:mm:ss)	实际速度 (cm/s)	实际速度 (km/h)	计算加速度 (m/s²)	计算距离(m)	计算公里标 (m)	当前坡度 (‰)	计算牵引力 (N)	计算牵引功率 (Kw)
0:00:00	0	0	0	0	13594	0	0	0
0:00:01	92.04189052	3.313508059	0.920418905	1.380628358	13592.61937	0	178836.6584	164.6046413
0:00:02	183.86236	6.619044959	0.918204695	3.678354305	13590.32165	0	178406.8785	328.0230972
0:00:03	275.4231955	9.915235038	0.915608355	6.890390437	13587.10961	0	177902.9273	489.9859271
0:00:04	366.6865964	13.20071747	0.912634009	11.01357341	13582.98643	0	177325.6045	650.2292235
0:00:05	457.6152358	16.47414849	0.909286394	16.04436896	13577.95563	0	176675.8301	808.4955165
0:00:06	548.172321	19.73420356	0.905570852	21.9788776	13572.02112	0	175954.6408	964.5346386
0:00:07	638.3216518	22.97957946	0.901493308	28.81284077	13565.18716	0	175163.1867	1118.104547
0:00:08	728.0276768	26.20899637	0.89706025	36.54164766	13557.45835	0	174302.727	1268.972094
0:00:09	817.2555477	29.42119972	0.892278709	45.16034249	13548.83966	0	173374.6264	1416.913752

0:00:10	905.9711708	32.61496215	0.887156231	54.66363232	13539.33637	0	172380.3498	1561.716273
0:00:11	994.1412565	35.78908524	0.881700858	65.04589531	13528.9541	0	171321.4581	1703.177296
0:00:12	1081.733366	38.94240119	0.875921097	76.30118952	13517.69881	0	170199.6025	1841.10589
0:00:13	1168.715956	42.07377441	0.869825895	88.42326203	13505.57674	0	169016.5195	1975.323031
0:00:14	1255.058417	45.182103	0.863424607	101.4055585	13492.59444	0	167774.025	2105.662022
0:00:15	1340.731113	48.26632008	0.856726969	115.2412331	13478.75877	0	166474.0087	2231.968831
0:00:16	1425.70542	51.32539512	0.849743066	129.9231588	13464.07684	0	165118.4281	2354.102379
0:00:17	1509.95375	54.358335	0.8424833	145.443938	13448.55606	0	163709.3025	2471.934752
0:00:18	1593.449586	57.3641851	0.834958361	161.795913	13432.20409	0	162248.7065	2585.351343
0:00:19	1658.244721	59.69680995	0.647951346	178.7023359	13415.29766	0	125915.6899	2087.990281
0:00:20	1716.656229	61.79962424	0.584115081	196.1609558	13397.83904	0	113513.8388	1948.642384
0:00:21	1769.855369	63.71479328	0.531991401	214.1255051	13379.87449	0	103387.5963	1829.810925
0:00:22	1818.708728	65.4735142	0.488533587	232.5568592	13361.44314	0	94945.0143	1726.773262
0:00:23	1863.873268	67.09943764	0.451645401	251.4214146	13342.57859	0	87778.81413	1636.085851
0:00:24	1849.81977	66.59351173	-0.140534974	269.8493448	13324.15066	0	0	0
0:00:25	1835.916891	66.09300807	-0.139028795	288.1389993	13305.861	0	0	0
0:00:26	1822.162123	65.59783643	-0.137547678	306.2918467	13287.70815	0	0	0
0:00:27	1806.789016	65.04440457	-0.153731071	324.2828713	13269.71713	1.8	0	0
0:00:28	1791.57768	64.49679647	-0.152113362	342.1225914	13251.87741	1.8	0	0
0:00:29	1776.525344	63.9549124	-0.150523352	359.8125832	13234.18742	1.8	0	0
0:00:30	1761.629303	63.41865489	-0.148960418	377.354396	13216.6456	1.8	0	0
0:00:31	1746.886908	62.88792867	-0.14742395	394.7495531	13199.25045	1.8	0	0
0:00:32	1732.295572	62.36264058	-0.145913359	411.9995522	13182.00045	1.8	0	0



0:00:33	1717.852765	61.84269953	-0.144428069	429.1058658	13164.89413	1.8	0	0
0:00:34	1703.556013	61.32801646	-0.142967522	446.0699421	13147.93006	1.8	0	0
0:00:35	1689.402895	60.81850423	-0.141531175	462.8932055	13131.10679	1.8	0	0
0:00:36	1675.391045	60.31407763	-0.140118499	479.5770567	13114.42294	1.8	0	0
0:00:37	1661.518147	59.8146533	-0.138728981	496.1228737	13097.87713	1.8	0	0
0:00:38	1647.781935	59.32014966	-0.137362121	512.532012	13081.46799	1.8	0	0
0:00:39	1634.180192	58.83048691	-0.136017432	528.8058052	13065.19419	1.8	0	0
0:00:40	1620.710748	58.34558693	-0.13469444	544.9455654	13049.05443	1.8	0	0
0:00:41	1607.371479	57.86537326	-0.133392685	560.9525839	13033.04742	1.8	0	0
0:00:42	1594.160308	57.38977107	-0.132111718	576.8281311	13017.17187	1.8	0	0
0:00:43	1581.075197	56.9187071	-0.130851103	592.5734575	13001.42654	1.8	0	0
0:00:44	1568.114156	56.45210962	-0.129610413	608.1897939	12985.81021	1.8	0	0
0:00:45	1555.275233	55.98990837	-0.128389235	623.6783516	12970.32165	1.8	0	0
0:00:46	1542.556516	55.53203458	-0.127187165	639.0403232	12954.95968	1.8	0	0
0:00:47	1529.956135	55.07842087	-0.126003809	654.2768826	12939.72312	1.8	0	0
0:00:48	1517.472257	54.62900124	-0.124838786	669.3891858	12924.61081	1.8	0	0
0:00:49	1505.103085	54.18371104	-0.123691721	684.3783708	12909.62163	1.8	0	0
0:00:50	1498.040859	53.92947094	-0.070622251	699.3234683	12894.67653	-3.5	0	0
0:00:51	1491.042807	53.67754105	-0.069980525	714.1989061	12879.80109	-3.5	0	0
0:00:52	1484.108118	53.42789225	-0.069346889	729.0053138	12864.99469	-3.5	0	0
0:00:53	1477.235997	53.1804959	-0.068721207	743.7433132	12850.25669	-3.5	0	0
0:00:54	1470.425662	52.93532385	-0.068103348	758.4135181	12835.58648	-3.5	0	0
0:00:55	1463.676344	52.69234839	-0.067493183	773.016535	12820.98347	-3.5	0	0

0:00:56	1456.987286	52.45154228	-0.066890585	787.5529625	12806.44704	-3.5	0	0
0:00:57	1450.357743	52.21287874	-0.06629543	802.0233922	12791.97661	-3.5	0	0
0:00:58	1443.786983	51.97633139	-0.065707597	816.4284083	12777.57159	-3.5	0	0
0:00:59	1437.274286	51.74187431	-0.065126967	830.7685877	12763.23141	-3.5	0	0
0:01:00	1430.818944	51.50948199	-0.064553423	845.0445004	12748.9555	-3.5	0	0
0:01:01	1424.420259	51.27912932	-0.063986851	859.2567095	12734.74329	-3.5	0	0
0:01:02	1418.077545	51.05079162	-0.063427139	873.4057714	12720.59423	-3.5	0	0
0:01:03	1411.790127	50.82444458	-0.062874179	887.4922356	12706.50776	-3.5	0	0
0:01:04	1405.557341	50.60006428	-0.062327861	901.5166451	12692.48335	-3.5	0	0
0:01:05	1399.378533	50.37762719	-0.061788081	915.4795364	12678.52046	-3.5	0	0
0:01:06	1393.253059	50.15711013	-0.061254737	929.3814396	12664.61856	-3.5	0	0
0:01:07	1387.180287	49.93849032	-0.060727725	943.2228786	12650.77712	-3.5	0	0
0:01:08	1381.159592	49.72174531	-0.060206948	957.0043711	12636.99563	-3.5	0	0
0:01:09	1375.190361	49.506853	-0.059692308	970.7264285	12623.27357	-3.5	0	0
0:01:10	1369.27199	49.29379165	-0.05918371	984.3895566	12609.61044	-3.5	0	0
0:01:11	1363.403884	49.08253983	-0.058681059	997.9942549	12596.00575	-3.5	0	0
0:01:12	1357.585458	48.87307648	-0.058184265	1011.541017	12582.45898	-3.5	0	0
0:01:13	1351.816134	48.66538082	-0.057693238	1025.030332	12568.96967	-3.5	0	0
0:01:14	1346.095345	48.45943242	-0.057207889	1038.462682	12555.53732	-3.5	0	0
0:01:15	1340.422532	48.25521115	-0.056728131	1051.838543	12542.16146	-3.5	0	0
0:01:16	1334.797144	48.05269718	-0.05625388	1065.158387	12528.84161	-3.5	0	0
0:01:17	1329.218639	47.851871	-0.055785052	1078.422681	12515.57732	-3.5	0	0
0:01:18	1323.686482	47.65271336	-0.055321566	1091.631885	12502.36811	-3.5	0	0

0:01:19	1318.200148	47.45520533	-0.054863342	1104.786455	12489.21354	-3.5	0	0
0:01:20	1312.759118	47.25932824	-0.0544103	1117.886841	12476.11316	-3.5	0	0
0:01:21	1307.362881	47.06506373	-0.053962364	1130.933489	12463.06651	-3.5	0	0
0:01:22	1302.010936	46.87239368	-0.053519458	1143.926838	12450.07316	-3.5	0	0
0:01:23	1296.702785	46.68130026	-0.053081508	1156.867325	12437.13267	-3.5	0	0
0:01:24	1291.437941	46.49176588	-0.052648439	1169.755381	12424.24462	-3.5	0	0
0:01:25	1286.215923	46.30377322	-0.052220182	1182.59143	12411.40857	-3.5	0	0
0:01:26	1281.036256	46.11730523	-0.051796664	1195.375894	12398.62411	-3.5	0	0
0:01:27	1275.898475	45.93234508	-0.051377819	1208.10919	12385.89081	-3.5	0	0
0:01:28	1270.802117	45.74887621	-0.050963576	1220.791729	12373.20827	-3.5	0	0
0:01:29	1265.74673	45.56688227	-0.050553871	1233.42392	12360.57608	-3.5	0	0
0:01:30	1260.731866	45.38634718	-0.050148637	1246.006164	12347.99384	-3.5	0	0
0:01:31	1255.757085	45.20725507	-0.04974781	1258.538861	12335.46114	-3.5	0	0
0:01:32	1250.821952	45.02959029	-0.049351328	1271.022405	12322.9776	-3.5	0	0
0:01:33	1169.032656	42.08517561	-0.817892966	1282.303785	12311.69622	-3.5	0	0
0:01:34	1087.877068	39.16357445	-0.811555877	1292.776778	12301.22322	-3.5	0	0
0:01:35	1007.31993	36.26351748	-0.805571381	1302.447191	12291.55281	-3.5	0	0
0:01:36	927.3269308	33.38376951	-0.799929993	1311.320495	12282.6795	-3.5	0	0
0:01:37	844.4346449	30.39964721	-0.828922859	1319.35038	12274.64962	0	0	0
0:01:38	762.0613202	27.43420753	-0.823733247	1326.559127	12267.44087	0	0	0
0:01:39	680.1724643	24.48620872	-0.818888559	1332.951407	12261.04859	0	0	0
0:01:40	598.7343569	21.55443685	-0.814381075	1338.53156	12255.46844	0	0	0
0:01:41	517.7139904	18.63770365	-0.810203665	1343.303598	12250.6964	0	0	0

0:01:42	437.0790137	15.73484449	-0.806349766	1347.271214	12246.72879	0	0	0
0:01:43	356.7976778	12.8447164	-0.802813359	1350.437784	12243.56222	0	0	0
0:01:44	276.8387831	9.966196191	-0.799588947	1352.806377	12241.19362	0	0	0
0:01:45	197.1716291	7.098178646	-0.79667154	1354.379758	12239.62024	0	0	0
0:01:46	117.7659655	4.239574757	-0.794056636	1355.160389	12238.83961	0	0	0
0:01:47	0	0	-0.791740207	1355.150438	12238.84956	0	0	0
0:01:48	0	0	0	1355.150438	12238.84956	0	4.038392166	0
0:01:49	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:50	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:51	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:52	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:53	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:54	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:55	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:56	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:57	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:58	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:01:59	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:00	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:01	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:02	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:03	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:04	0	0	0	1355.150438	12238.84956	0	3.867208821	0

0:02:05	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:06	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:07	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:08	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:09	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:10	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:11	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:12	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:13	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:14	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:15	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:16	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:17	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:18	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:19	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:20	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:21	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:22	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:23	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:24	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:25	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:26	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:27	0	0	0	1355.150438	12238.84956	0	3.867208821	0

0:02:28	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:29	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:30	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:31	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:32	0	0	0	1355.150438	12238.84956	0	3.867208821	0
0:02:33	81.59386046	2.937378977	0.815938605	1356.374346	12237.62565	0	158536.6584	129.3561798
0:02:34	162.9933907	5.867762066	0.813995303	1358.411278	12235.58872	0	158159.4621	257.78947
0:02:35	244.168604	8.790069746	0.811752133	1361.25884	12232.74116	0	157724.0614	385.1126388
0:02:36	325.0898309	11.70323391	0.809212269	1364.914344	12229.08566	0	157231.0719	511.1422258
0:02:37	405.727762	14.60619943	0.806379311	1369.374812	12224.62519	0	156681.1927	635.6990968
0:02:38	486.0534898	17.49792563	0.803257278	1374.636975	12219.36302	0	156075.204	758.6089757
0:02:39	566.0385494	20.37738778	0.799850596	1380.697286	12213.30271	0	155413.9646	879.7029506
0:02:40	645.6549576	23.24357847	0.796164082	1387.551918	12206.44808	0	154698.4095	998.8179505
0:02:41	724.8752507	26.09550902	0.792202931	1395.196772	12198.80323	0	153929.5474	1115.797192
0:02:42	803.6725209	28.93221075	0.787972702	1403.627483	12190.37252	0	153108.4569	1230.490595
0:02:43	882.0204507	31.75273623	0.783479298	1412.839427	12181.16057	0	152236.2842	1342.75516
0:02:44	959.8933461	34.55616046	0.778728954	1422.827725	12171.17227	0	151314.239	1452.455312
0:02:45	1037.266167	37.34158202	0.773728212	1433.587251	12160.41275	0	150343.5915	1559.463209
0:02:46	1114.114558	40.1081241	0.768483909	1445.112639	12148.88736	0	149325.6685	1663.659012
0:02:47	1190.414873	42.85493544	0.763003151	1457.398289	12136.60171	0	148261.8496	1764.931109
0:02:48	1266.144203	45.58119132	0.757293301	1470.438378	12123.56162	0	147153.5635	1863.176315
0:02:49	1341.280398	48.28609434	0.751361949	1484.226863	12109.77314	0	146002.2841	1958.300017
0:02:50	1415.802088	50.96887519	0.745216901	1498.757492	12095.24251	0	144809.5259	2050.216293

0:02:51	1489.688704	53.62879333	0.738866151	1514.023812	12079.97619	0	143576.8409	2138.847979
0:02:52	1562.92049	56.26513764	0.732317863	1530.019176	12063.98082	0	142305.8135	2224.126718
0:02:53	1622.176976	58.39837115	0.592564866	1546.537228	12047.46277	0	115153.8139	1867.998657
0:02:54	1675.895955	60.33225439	0.537189787	1563.564782	12030.43522	0	104395.8072	1749.565111
0:02:55	1725.025328	62.10091181	0.49129373	1581.060683	12012.93932	0	95479.45188	1647.044728
0:02:56	1712.420162	61.64712584	-0.126051659	1598.121858	11995.87814	0	0	0
0:02:57	1697.0021	61.0920756	-0.154180621	1615.014789	11978.98521	3	0	0
0:02:58	1681.738514	60.5425865	-0.152635861	1631.755856	11962.24414	3	0	0
0:02:59	1666.626781	59.99856413	-0.151117326	1648.346565	11945.65343	3	0	0
0:03:00	1651.664338	59.45991617	-0.149624434	1664.788397	11929.2116	3	0	0
0:03:01	1636.848676	58.92655235	-0.148156617	1681.082805	11912.91719	3	0	0
0:03:02	1622.177344	58.39838438	-0.146713324	1697.231222	11896.76878	3	0	0
0:03:03	1607.647942	57.87532591	-0.14529402	1713.235054	11880.76495	3	0	0
0:03:04	1593.258124	57.35729245	-0.143898184	1729.095686	11864.90431	3	0	0
0:03:05	1579.005593	56.84420134	-0.142525308	1744.81448	11849.18552	3	0	0
0:03:06	1564.888103	56.3359717	-0.141174899	1760.392773	11833.60723	3	0	0
0:03:07	1550.903455	55.83252438	-0.139846477	1775.831885	11818.16812	3	0	0
0:03:08	1537.049498	55.33378191	-0.138539575	1791.13311	11802.86689	3	0	0
0:03:09	1523.324124	54.83966846	-0.137253737	1806.297724	11787.70228	3	0	0
0:03:10	1509.725272	54.35010978	-0.135988521	1821.326983	11772.67302	3	0	0
0:03:11	1496.250922	53.86503319	-0.134743496	1836.22212	11757.77788	3	0	0
0:03:12	1482.899098	53.38436752	-0.133518242	1850.984352	11743.01565	3	0	0
0:03:13	1469.667863	52.90804307	-0.132312349	1865.614874	11728.38513	3	0	0

0:03:14	1456.555321	52.43599155	-0.13112542	1880.114865	11713.88514	3	0	0
0:03:15	1443.559614	51.96814611	-0.129957067	1894.485482	11699.51452	3	0	0
0:03:16	1430.678923	51.50444123	-0.12880691	1908.727868	11685.27213	3	0	0
0:03:17	1417.911465	51.04481274	-0.127674583	1922.843146	11671.15685	3	0	0
0:03:18	1405.255492	50.58919773	-0.126559724	1936.832421	11657.16758	3	0	0
0:03:19	1392.709294	50.13753458	-0.125461986	1950.696783	11643.30322	3	0	0
0:03:20	1380.271191	49.68976289	-0.124381025	1964.437304	11629.5627	3	0	0
0:03:21	1367.93954	49.24582345	-0.12331651	1978.055041	11615.94496	3	0	0
0:03:22	1355.712729	48.80565824	-0.122268115	1991.551034	11602.44897	3	0	0
0:03:23	1343.589176	48.36921035	-0.121235525	2004.926308	11589.07369	3	0	0
0:03:24	1336.467333	48.11282399	-0.071218431	2018.255372	11575.74463	-2	0	0
0:03:25	1329.404924	47.85857725	-0.070624096	2031.51411	11562.48589	-2	0	0
0:03:26	1322.401221	47.60644397	-0.070037021	2044.703103	11549.2969	-2	0	0
0:03:27	1315.455513	47.35639846	-0.069457086	2057.82293	11536.17707	-2	0	0
0:03:28	1308.567095	47.10841543	-0.068884177	2070.874159	11523.12584	-2	0	0
0:03:29	1301.735277	46.86246998	-0.06831818	2083.857353	11510.14265	-2	0	0
0:03:30	1294.959379	46.61853763	-0.067758986	2096.773067	11497.22693	-2	0	0
0:03:31	1288.23873	46.37659428	-0.067206485	2109.621851	11484.37815	-2	0	0
0:03:32	1281.572673	46.13661623	-0.06666057	2122.404247	11471.59575	-2	0	0
0:03:33	1274.960559	45.89858014	-0.066121138	2135.120792	11458.87921	-2	0	0
0:03:34	1268.401751	45.66246303	-0.065588086	2147.772016	11446.22798	-2	0	0
0:03:35	1261.895619	45.4282423	-0.065061314	2160.358441	11433.64156	-2	0	0
0:03:36	1255.441547	45.19589569	-0.064540723	2172.880586	11421.11941	-2	0	0



0:03:37	1249.038925	44.96540131	-0.064026218	2185.338963	11408.66104	-2	0	0
0:03:38	1242.687155	44.73673757	-0.063517704	2197.734075	11396.26592	-2	0	0
0:03:39	1236.385646	44.50988326	-0.063015087	2210.066424	11383.93358	-2	0	0
0:03:40	1230.133818	44.28481746	-0.062518278	2222.336503	11371.6635	-2	0	0
0:03:41	1223.9311	44.06151959	-0.062027187	2234.544801	11359.4552	-2	0	0
0:03:42	1217.776927	43.83996937	-0.061541726	2246.691799	11347.3082	-2	0	0
0:03:43	1211.670746	43.62014686	-0.06106181	2258.777976	11335.22202	-2	0	0
0:03:44	1205.612011	43.40203238	-0.060587355	2270.803802	11323.1962	-2	0	0
0:03:45	1199.600183	43.18560658	-0.060118278	2282.769745	11311.23026	-2	0	0
0:03:46	1193.634733	42.97085039	-0.059654497	2294.676265	11299.32374	-2	0	0
0:03:47	1187.71514	42.75774502	-0.059195935	2306.523818	11287.47618	-2	0	0
0:03:48	1181.840888	42.54627198	-0.058742512	2318.312856	11275.68714	-2	0	0
0:03:49	1176.011473	42.33641303	-0.058294152	2330.043824	11263.95618	-2	0	0
0:03:50	1170.226395	42.12815022	-0.05785078	2341.717162	11252.28284	-2	0	0
0:03:51	1164.485163	41.92146586	-0.057412323	2353.333308	11240.66669	-2	0	0
0:03:52	1158.787292	41.71634252	-0.056978707	2364.892691	11229.10731	-2	0	0
0:03:53	1153.132306	41.51276301	-0.056549862	2376.395739	11217.60426	-2	0	0
0:03:54	1147.519734	41.31071043	-0.056125718	2387.842874	11206.15713	-2	0	0
0:03:55	1141.949113	41.11016808	-0.055706207	2399.234512	11194.76549	-2	0	0
0:03:56	1136.419987	40.91111954	-0.055291261	2410.571066	11183.42893	-2	0	0
0:03:57	1130.931906	40.71354861	-0.054880815	2421.852945	11172.14706	-2	0	0
0:03:58	1125.484425	40.51743932	-0.054474803	2433.080552	11160.91945	-2	0	0
0:03:59	1120.077109	40.32277593	-0.054073162	2444.254286	11149.74571	-2	0	0

0:04:00	1114.709526	40.12954295	-0.053675829	2455.374543	11138.62546	-2	0	0
0:04:01	1109.381252	39.93772507	-0.053282743	2466.441715	11127.55829	-2	0	0
0:04:02	1104.091868	39.74730724	-0.052893844	2477.456186	11116.54381	-2	0	0
0:04:03	1098.84096	39.55827457	-0.052509073	2488.418341	11105.58166	-2	0	0
0:04:04	1093.628123	39.37061244	-0.05212837	2499.328558	11094.67144	-2	0	0
0:04:05	1088.452955	39.1843064	-0.05175168	2510.187212	11083.81279	-2	0	0
0:04:06	1083.315061	38.99934219	-0.051378945	2520.994673	11073.00533	-2	0	0
0:04:07	1078.21405	38.81570579	-0.051010111	2531.751309	11062.24869	-2	0	0
0:04:08	1073.149537	38.63338335	-0.050645123	2542.457481	11051.54252	-2	0	0
0:04:09	1068.121145	38.4523612	-0.050283929	2553.113551	11040.88645	-2	0	0
0:04:10	1063.128497	38.27262589	-0.049926475	2563.719873	11030.28013	-2	0	0
0:04:11	1056.211226	38.02360413	-0.069172711	2574.247399	11019.7526	0	0	0
0:04:12	1049.34278	37.77634006	-0.068684464	2584.706484	11009.29352	0	0	0
0:04:13	957.0855028	34.4550781	-0.922572767	2593.816053	11000.18395	0	0	0
0:04:14	865.4555053	31.15639819	-0.916299975	2602.012458	10991.98754	0	0	0
0:04:15	774.4098515	27.87875465	-0.910456537	2609.301328	10984.69867	0	0	0
0:04:16	683.9066445	24.6206392	-0.90503207	2615.687879	10978.31212	0	0	0
0:04:17	593.9049431	21.38057795	-0.900017014	2621.176919	10972.82308	0	0	0
0:04:18	504.3646838	18.15712862	-0.895402593	2625.772865	10968.22713	0	0	0
0:04:19	415.2466054	14.94887779	-0.891180785	2629.479741	10964.52026	0	0	0
0:04:20	326.5121763	11.75443835	-0.887344291	2632.30119	10961.69881	0	0	0
0:04:21	238.1235256	8.57244692	-0.883886507	2634.240482	10959.75952	0	0	0
0:04:22	150.0433755	5.401561517	-0.880801501	2635.300515	10958.69948	0	0	0

0:04:23	62.23497667	2.24045916	-0.878083988	2635.483823	10958.51618	0	0	0
0:04:24	-25.33795495	-0.912166378	-0.875729316	2634.792579	10959.20742	0	0	0
0:04:25	-112.7112995	-4.05760678	-0.873733445	2633.228599	10960.7714	0	0	0

## 10.2 主要程序

### 1、 gumcmD\_01.m

```
clear;clc;
O = 13594;
T = 111;
L = O-12240;
M = 194.295;
record = [];
a = [];
v = [];
err = 100000000000;
for u1 = 0.5:0.1:1
    for u2 = 0.2:0.1:1
        for t1 = 5:50
            for t2 = t1+1:109
                v(1) = 0;%初速度为 0
                l(1) = 0;
                a(1)=0;
                ww(1)=0;
                for t = 2:T
                    %计算阻力 (阻力=基本阻力w0+坡道阻力wi+曲线阻力wc)
                    w0 = W0(v(t-1));
                    wi = Wi(l(t-1));
                    wc = 0;
                    w = (w0+wi+wc)*9.8*M/1000; %KN
                    ww(t) = w;
                    %计算最大牵引力
                    F_max = F_MAX(v(t-1)); %KN
                    %计算实际牵引力
                    F_real = u1 * F_max;

                    %% 加速阶段
                    if t<=t1
                        %计算加速度
                        a(t) = (F_real-w)/M;
                        if a(t) > 1
                            a(t) = 1;
                        end

                    %% 惰行阶段
                    elseif t>t1 && t<=t2
                        a(t) = (-w)/M;

                    %% 减速阶段
                    elseif t>t2
                        %计算最大制动
                        b_max = u2*B_max(v(t-1)); %KN
                        %计算加速度
                        a(t) = (-b_max-w)/M; %m/s
                    End

                    %%
                    %速度改变
                    v(t) =v(t-1)+a(t); %m/s
                    %计算路程
```

```

        s(t) = v(t)+0.5*a(t);      %m
        l(t) = l(t-1)+s(t);      %m
    end
    %计算耗能
    F = [];
    E = 0;
    for i=1:length(a)
        if a(i)>=0
            F(i) = ww(i)+M*1000*a(i);
        else
            F(i) = 0;
        end
        E = E+F(i)*s(i);
    end
    %路程、速度约束
    if abs(l(end)-1354) + abs(v(end)) < 0.5
        if E < err
            err = E;
            record = [u1,u2,t1,t2,E];
        end
    end
end
end
end
end
end
end

```

## 2、gumcmD\_02main.m

```

clear;clc;
global dis_init
%每站路的公里标
sign =
[22903,21569,20283,18197,15932,13594,12240,10960,9422,8429,6447,4081,2806,175];
%每一段的公里数
dis_every = abs(diff(sign));
%每一段起点的初始里程
dis_init = [0,dis_every(1:end-1)];
%给每一段分配时间
tt = [100 96 156 169 174 101 95 115 74 148 177 95 196];
%计算不同路段的最优参数以及最低能耗
data = [];
for i = 1:length(sign)
    temp = gumcmD_02(i,tt(i))
    data = [data;temp];
end

```

## 3、gumcmD\_02.m

```

function record = gumcmD_02(n,time)
%n为第几段, n=1...13
%time为跑完这一段规定的时间
global dis_init
T = time+1;
M = 194.295;%列车的质量 ( 吨 )
record = [];
a = zeros(1,T);
v = zeros(1,T);
s = zeros(1,T);
l = zeros(1,T);

```

```

ww = zeros(1,T);
err = 10000000000;
wucha = 1;

flag = -1;
while flag < 0
    for u1 = 0.7:0.1:1
        for u2 = 0.4:0.1:1
            for t1 = 5:floor(T/2)
                for t2 = t1+1:T
                    v(1) = 0;%初始速度为 0
                    l(1) = sum(dis_init(1:n));%初始路程
                    a(1)=0;%初始加速度为 0
                    ww(1)=0;

                    for t = 2:T
                        %计算阻力 (阻力=基本阻力w0+坡道阻力wi+曲线阻力wc)
                        w0 = W0(v(t-1));
                        wi = Wi(l(t-1));
                        wc = Wc(l(t-1));
                        w = (w0+wi+wc)*9.8*M/1000;%KN
                        ww(t) = w;
                        %计算最大牵引力
                        F_max = F_MAX(v(t-1));%KN
                        %计算实际牵引力
                        F_real = u1 * F_max;
                        %% 加速阶段
                        if t <= t1
                            %计算加速度
                            a(t) = (F_real-w)/M;
                            if a(t) > 1
                                a(t) = 1;
                            end
                            %% 惰行阶段
                        elseif t > t1 && t <= t2
                            a(t) = (-w)/M;
                            %% 减速阶段
                        elseif t > t2
                            %计算最大制动
                            b_max = u2*B_max(v(t-1));%KN
                            %计算加速度
                            a(t) = (-b_max-w)/M; %m/s
                        end
                        %%
                        %速度改变
                        v(t) = v(t-1)+a(t); %m/s
                        %计算路程
                        s(t) = v(t)+0.5*a(t); %m
                        l(t) = l(t-1)+s(t); %m
                    end
                    %计算耗能
                    F = [];
                    E = 0;
                    for i=1:length(a)
                        if a(i) >= 0
                            F(i) = ww(i)+M*1000*a(i);
                        else
                            F(i) = 0;
                        end
                    end
                end
            end
        end
    end
    flag = 1;
end

```

```

        end
        E = E+F(i)*s(i);
    end
    %路程、速度约束
    if max(v) <= 80/3.6
        if abs(l(end)-2634) + abs(v(end)) < wucha
            if E < err
                err = E;
                record = [u1,u2,t1,t2,E,wucha];%输出
            end
        end
    end
end
end
end
end
end
end
if isempty(record)
    wucha = wucha+1
else
    flag = 1;
end
end
end
end

```

#### 4、F\_MAX.m

```

function f = F_MAX(v)
v=v*3.6;
if v<=55
    f = 203;
else
    f = -0.002032*v^3+0.4928*v^2-42.13*v+ 1343;
end

```

#### 5、B\_MAX.m

```

function b = B_max(v)
v=v*3.6;
if v<=77
    b = 166;
else
    b = 0.1343*v^2-25.07*v+ 1300;
end

```

#### 6、W0.m

```

function y = W0(v)
v=v*3.6;
y = 2.031+0.0622*v+0.001807*v^2;

```

#### 7、Wi.m

```

%坡道附加阻力
function f = Wi(l)
load podao
m = 22903-l;
%找到对应的区间
for i = 1:length(podao)
    if m >= podao(i,1)&& m <= podao(i,3)

```

```

        break;
    end
end
f = -podao(i,2);

```

#### 8、Wc.m

```

%曲率附加阻力
function f = Wc(l)
load qulv
m = 22903-l;
%找到对应的区间
for i = 1:length(qulv)
    if m >= qulv(i,1) && m <= qulv(i,3)
        break;
    end
end
R = qulv(i,2);
if R < 1
    f = 0;
else
    f = 600/R;
end

```